IBM InfoSphere Optim for z/OS
Version 11 Release 3

Batch Utilities

IBM

IBM InfoSphere Optim for z/OS
Version 11 Release 3

Batch Utilities

Note

Before using this information and the product it supports, read the information in “Notices” on page 163.

Version 11 Release 3

This edition applies to version 11, release 3 of IBM InfoSphere® Optim for z/OS and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this publication

This document provides information on the Optim™ Batch Utility, which you can use to maintain the Optim environment or automate Optim processes.

Chapter 1. About this Manual

IBM® Optim for z/OS® (hereinafter referred to as Optim) includes the components Access, Archive, Compare, and Move. This manual provides information on the Optim Batch Utility, which you can use to maintain the Optim environment or automate Optim processes.

Organization

The information in this manual is organized as follows:

1. Introduction

   General information helpful in using the Optim maintenance and processing Batch Utility control statements.
2. Maintenance Utilities

   Describes control statements used to:
   - Retrieve information from the Optim Directory
   - Maintain Optim objects in the Directory
   - Manage Archive Files, Archive File entries, and Archive log entries
   - Migrate Optim objects.
3. Processing Utilities

   Describes control statements used to run Optim processes (such as Extract, Archive, Insert, Convert, Compare, Delete, Search, and Restore) in batch mode.
4. File Allocation Parameters

   Describes how to allocate a file, specifying the file name and optional parameters, such as block size and processing mode.
5. AGING Keyword Parameters

   Describes parameters to be used in the Aging process.
6. Definition Keyword Parameters

   Describes the parameters that make up the following definition keywords:
   - ACCESS_DEFINITION_DEFINE
   - COMPARE_DEFINITION_DEFINE
   - TABLE_MAP_DEFINE.

Documentation

The following is a list of the Optim documentation for the z/OS environment.

Optim for z/OS Access User Manual
Used in conjunction with the *Common Elements Manual*, this user manual provides detailed information on using Access to access and edit a set of related data, and create and maintain Access Definitions, primary keys, and relationships in the Optim Directory.

Optim for z/OS Access Introduction
Overview of Access and a demonstration of key product features using scenarios based on the sample database distributed with Optim.

Optim for z/OS Archive User Manual
Used in conjunction with the *Common Elements Manual*, this user manual provides detailed information on using the Archive facilities to archive and restore sets of related data.

Optim for z/OS Compare User Manual

Used in conjunction with the *Common Elements Manual*, this user manual provides detailed information on using the Compare facilities to compare two sets of related data.

Optim for z/OS Compare for IMS, VSAM and Sequential Files

Used in conjunction with the *Common Elements Manual*, this user manual provides detailed information on comparing two sets of data from IMS, VSAM or sequential files.

Optim for z/OS Compare Introduction

Overview of Compare and a demonstration of key product features using scenarios based on the sample database distributed with Optim.

Optim for z/OS Move User Manual

Used in conjunction with the *Common Elements Manual*, this user manual provides detailed information on using Move for migrating sets of related data.

Optim for z/OS Move Introduction

Overview of Move and a demonstration of key product features using scenarios based on the sample database distributed with Optim.

Optim for z/OS Command Reference Manual

Detailed information about the commands for Optim.

Optim for z/OS Common Elements Manual

Detailed information about creating and maintaining the objects common to Optim, i.e., Access Definitions, Optim Primary Keys and Relationships, Table Maps, and information needed to export and import these objects from one IBM DB2® subsystem to another. This manual also provides information needed to:

- Convert or mask data in Archive and Extract Files.
- Restart or retry an Insert, Delete, or Restore Process that fails for some reason.
- Browse data in Archive, Extract, Compare, and Control Files or select specific rows from the Start Table for an Archive, Restore, Extract, or Compare Process using Point-and-Shoot.

Optim for z/OS Customization Guide

Instructions for installing and maintaining Optim.

Optim for z/OS Batch Utilities Guide

Description of Batch Utility control statements used to maintain the Optim environment and to automate Optim processes.

Chapter 2. Introduction to Batch Utilities

The Optim Batch Utility control statements help you maintain the Optim environment and automate processing.

You can create a Batch Utility processing job, using the Optim online panels and choosing to run the job in batch. Optim generates a series of Batch Utility control statements defining the function to be performed. Using ISPF or another editor, you can view and modify the specific job parameters directly. You must create Batch Utility maintenance jobs in an editor; you cannot generate them online. Also, certain keywords can only be entered using an editor.

Note: This document assumes you are familiar with the Optim online processes described in the *Common Elements Manual* and the User Manuals for the Optim components.

Maintenance Utilities

The maintenance utilities perform the following functions:
- Retrieve information from the Optim Directory. See Informational Control Statements.
- Maintain Optim objects in the Directory. See Utilities to Maintain Optim Objects.
- Manage Archive Files, Archive File entries, and Archive log entries. See Archive Entry and File Management.
- Migrate Optim objects. See Utilities to Migrate Optim Objects.

Processing Utilities

The processing utilities allow you to run Optim processes (such as Extract, Archive, Insert, Convert, Compare, Delete, Search, and Restore) in batch mode. Processing Utilities describes the processing utilities, their control statement syntax, and keywords.

Note: An object or archive security exit enabled from the Site Options panel is called during execution of the Batch Utility, when appropriate. If you use a security exit distributed with Optim, objects or Archive Files designated as PRIVATE cannot be accessed for any purpose, and objects or Archive Files designated as READONLY cannot be modified or deleted using the Batch Utility.

JCL Requirements

The JCL needed to execute the Batch Utility follows:

```
//         EXEC PGM=FOPMAIN,REGION=0M,
//            PARM=' [ CON UTILITY subsys planname sqlid userid |
                               BATCH subsys planname sqlid userid ] '

//SYSPRINT DD SYSOUT=*                         Diagnostic messages
//SYSTERM  DD SYSOUT=*                         Error and diagnostic messages
//PSDFDFLT DD DSN=your.loadlib(FOPMDFLT)       Default Site Options (optional)
//            DISP=SHR
//PSTRACE  DD SYSOUT=*                         Trace output, if Trace option activated
//PSDFASUM DD SYSOUT=*                         Summary of results of utility operations
//PSDFADIR DD SYSOUT=*                         Generated listings (DIR, COPY, RENAME, DELETE, EXPORT,
                                               or IMPORT statement)

//PSDFCERR DD SYSOUT=*                         Error report for Convert process
//PSDFDARL DD DSN=data.ace.rl.file             Record Layout PDS or file (If required)
//            DISP=SHR
//PSDFRPRT DD SYSOUT=*                         Report listings (REPORT, ARCHIVE, SEARCH, RESTORE,
                                                 or EXTRACT statement)
//PSDFEXPT DD DSN=export.file.dsn               Exported definitions (EXPORT statement)
//               DISP=SHR
//PSDFIMPT DD DSN=import.file.dsn               Definitions data set (IMPORT statement)
//               DISP=SHR
//PSDFEXTR DD DSN=extract.file.name             Extract or Archive file for process
//PSDFIDX  DD DSN=index.file.name               Archive Index file for process
//PSDFASUB DD DSN=subset.file.name              Subset file for process
//PSDFADUP DD DSN=duplicate.archive.file.name   Duplicate Archive file for process
//PSDFIDX2 DD DSN=dup.index.file.name           Duplicate Archive Index file for process
//PSDFCTRL DD DSN=control.file.name             Control File for process
//PSDFCNVT DD DSN=convert.file.name             Results file from Convert process
//PSDFCOMP DD DSN=compare.file.name             Results file from Compare process
//         DD *                                 Batch Utility control statements
//SYSIN  control statements
//PSDFOVRD DD *                                 File to provide batch overrides (optional)
         batch overrides
/*
```

A sample of this JCL is provided in the install library as member FOPJOBUT.

To use a custom module for default site options, replace member name FOPMDFLT in the PSDFDFLT DD statement with the member name of the custom site options module.

Notes:

- If you will be using a load or unload program, you must include the appropriate DD statements for that program. Refer to the program's documentation for the required DD statements.
- You may optionally supply a PSDFOVRD DD file, containing batch overrides for selected keywords. These overrides supersede the values specified in the equivalent Batch Utility keywords.

Configuring batch jobs for use with Optim data privacy providers

When building batch jobs through the Optim for z/OS interface that invoke the Optim data privacy providers, Optim automatically adds required parameters to the produced JCL. However, if you are manually creating an Optim for z/OS batch job that invokes the Optim data privacy providers, you must make the following modification to your JCL.

1. Verify you have performed the licensing and environment variable file steps. Refer to the *Customization Guide* section on installing the data privacy providers.
2. When you run a program directly under TSO that depends on the UNIX System Services environment and settings, you must:
   - specify to the Language Environment® the location of the UNIX System Services environment variables that the data privacy providers require
   - tell the Language Environment to enable POSIX to allow it to access the UNIX System Services environment

The following is an example of a batch job (JOB1) and program (IOQXXXX) that points to the necessary environment variables through the ENVAR parameter. The _CEE_ENVFILE_S parameter identifies FOPENV as the DD statement that specifies the file that contains the actual environment setting statements, and it is specified with the CEEOPTS DD.

```
//JOB1 EXEC PGM=IOQXXXX
//CEEOPTS DD *
ENVAR(_CEE_ENVFILE_S=DD:FOPENV),POSIX(ON)
/*
//FOPENV DD DISP=SHR,DSN=OPTIM.ODPP.ENV(V1130)
```

Change OPTIM.ODPP.ENV(V1130) to your environment variable file.

Converting Pre-Release 5.5 Batch Jobs

To convert pre-Release 5.5 batch jobs to the Batch Utility format, execute the Batch Utility, supplying one or more JCL_REFORMAT statements in the SYSIN data set. The JCL_REFORMAT statement converts batch jobs generated prior to Release 5.5 to the Batch Utility format. For more detailed information on converting existing batch jobs, see **Executing Batch Jobs Saved before Release 5.5** in the *Customization Guide*.

PSDFASUM, PSDFADIR, PSDFEXPT, PSDFIMPT

You can direct listings for PSDFASUM, PSDFADIR, PSDFEXPT, or PSDFIMPT to a data set, rather than to SYSOUT.

Required DCB attributes are:

RECFM=FB

For PSDFASUM and PSDFADIR:

LRECL=132

For PSDFEXPT and PSDFIMPT:

LRECL=80

The block size must be a multiple of the record length (i.e., 132 or 80). The data set may be sequential or partitioned. Note that, unlike a listing directed to SYSOUT, there is no carriage control byte before each line.

PSDFRPRT

If PSDFRPRT is assigned to SYSOUT, you can optionally specify DCB=LRECL=*n* on the DD statement to wrap print lines produced by the REPORT or SEARCH statement at *n* bytes.

If PSDFRPRT is assigned to a data set rather than SYSOUT, you must specify the following data set attribute:

RECFM=VB
or
RECFM=FB

If RECFM=VB is used, specify LRECL=136 and a BLKSIZE value of your choice.

If RECFM=FB is used, specify LRECL=132 and a BLKSIZE value that is a multiple of 132.

For a REPORT statement, whether RECFM=FB or RECFM=VB is used, do not specify values for LRECL and BLKSIZE. Optim will calculate the correct values at run time.

Wrapping of print lines produced by the REPORT or SEARCH statement is controlled by the WRAP_LINE keyword in the REPORT or SEARCH statement.

When PSDFRPRT is assigned to a data set, only one REPORT statement may be present for each execution of the Batch Utility. If more than one REPORT statement is present, only the last report will appear in the data set.

SYSIN DD * Statement

Control statements to carry out various functions are inserted after the SYSIN DD * statement.

Return Codes

The Batch Utility passes one of the following return codes to z/OS.

Code	Description
0	No errors or warnings
4	Warnings, but no errors
12	Errors

Control Statement Conventions

Use the following guidelines to format control statements:

- Begin a control statement in any column.
- Separate keywords and values or operands with one or more blanks.
- Enclose a list specified as an operand in parentheses and separate items in the list with commas.

  Note: You must leave a space after a comma that precedes a numeric value if the DB2® setup specifies a comma as the decimal point value.
- Continue a statement on a new line, starting in any column, by breaking the first line at any blank or special character (unless the blank or special character is inside a quoted literal string). A continuation indicator is not needed under these conditions.

  If there is insufficient room on a line to complete a literal string or Long Object Name (LON) (such as a table name or Creator ID), do the following:
  - Type a hyphen (-) in the right-most byte of Line 1 to indicate that the LON or string is continued on the next line. (The line can be broken with a hyphen at any point up to and including byte 80, and then continued on the next line.)
  - Continue the LON or string on Line 2, beginning in Column 1, as shown in the following example:

    ```
    REF_TABLE  PAYROLLDEPT.CUSTOMERS_VENDOR_SUP-
    PLY_INVENTORY
    ```

    In the example, a hyphen appears at the end of Line 1 after the P in “SUP”, and the LON continues on Line 2 with the P in “PLY.” The values on Lines 1 and 2 are concatenated to form the word SUPPLY.
- Indicate a comment line by placing hyphens or minus signs in Columns 1 and 2.
- End-statement indicators are not needed. A valid command verb, such as DELETE or DIRECTORY, indicates the end of a control statement and the beginning of the next. The normal end-file flag from the operating system indicates the end of the final statement.
- A keyword must be appropriate for the specified TYPE. Using a TYPE-specific keyword with an inappropriate TYPE causes an error condition, unless the IGNORE_TYPE keyword is used. (IGNORE_TYPE is valid only for the DIRECTORY or EXPORT statement.)
- An ERROR occurs if indicated objects are not in the Optim Directory.

Syntax

The control statements described in this document use the following syntax conventions:

KEYWORD
Statement keywords are shown in uppercase and must be supplied as shown.

text Variable text, describing parameter values, is shown in lowercase italics. The following descriptors indicate whether explicit values and/or patterns may be entered.

explicit filename
An explicit, fully qualified file name is required.

filename
An explicit, fully qualified file name or pattern is allowed.

objectname
An explicit object name or pattern is allowed.

group.user.name
An explicit Definition name is required (e.g., Access Definition, Compare Definition).

adname
An explicit Access Definition name or pattern is allowed.

mapid.name
An explicit Table Map name or pattern is allowed.

( ) Required as a statement delimiter. May group a series of qualifiers for a parameter. For example, (*text1, text2*,...) or (KEYWORD1 *text1*, KEYWORD2 *text2*,...).

[ ] Indicates an optional parameter.

{ } Indicates a choice of two or more settings from which one (and only one) must be selected.

< > Indicates a choice of two or more settings from which one or more may be selected.

| Separates options.

underscore
Indicates the default setting.

General Control Statements

You can use the following general Batch Utility control statements with any of the maintenance or processing control statements: **PAGESIZE**, **ERROR**, and **PASSWORD**.

PAGESIZE

A PAGESIZE statement governs the number of printed lines per page, excluding beginning and ending margins.

```
PAGESIZE [ nn ]
```

Each PAGESIZE statement applies until another PAGESIZE statement is encountered. The default operand for a PAGESIZE statement is 56; if no PAGESIZE statement applies, the page size is 56 lines plus beginning and ending margins. The following is an example of a PAGESIZE statement:

```
PAGESIZE  75
```

ERROR

An ERROR statement governs the handling of error conditions during a run.

```
ERROR { STOP | CONT }
```

Each ERROR statement applies until another ERROR statement is found. Keywords for an ERROR statement are:

STOP Halt the run if an error occurs; do not execute the next statement.

CONT
Continue the run, executing the next control statement, regardless of errors.

Note: When used for an IMPORT process, the ERROR CONT parameter must precede all other parameters.

If you do not provide an error statement, the run stops when an error occurs.

PASSWORD

A PASSWORD statement specifies a valid password that matches the administrator password in the site defaults module.

```
PASSWORD password
```

The PASSWORD statement can appear anywhere in the SYSIN file; however, it must precede any IMPORT statements that require administrator status for proper execution. For example, only an administrator can use the OWNERMODE keyword on the IMPORT statement to specify that the owner of imported objects is the original owner from the input file or an explicit owner. Therefore, the PASSWORD statement must precede the IMPORT statement if the OWNERMODE keyword is SOURCE or EXPLICIT. For an example of PASSWORD statement usage with the IMPORT statement, see IMPORT.

For security purposes, the password value is not displayed in the input listing. However, if a site does not want to expose the administrator password in the input queue, it can use a concatenation of data sets. Place the PASSWORD statement in the first data set, which is either password-protected or protected from public view by a security package. This data set would be followed by another data set (usually a SYSIN DD * data set) containing the other control statements.

Chapter 3. Maintenance Utilities

Maintenance utilities are an efficient means of monitoring the usage of Optim objects in order to ensure maximum benefit from Optim. For example, if a database table is removed or renamed, you can use a Batch Utility control statement to generate an impact analysis report of Optim objects that reference the table or column.

This report can be a useful tool to help you decide whether to remove and replace, or edit the affected Optim objects. If you decide to remove the objects, another Batch Utility control statement allows you to delete the same Optim objects from the Directory simply by changing the command verb in the control statement from DIRECTORY to DELETE. You can also use the Batch Utility to audit compliance with naming standards for Optim objects, review ownership of the objects, or to monitor Archive activities. Other maintenance utilities allow you to copy Optim objects in a Directory, and migrate Optim objects across directories.

There are four categories of maintenance utilities. There are utilities to

- Retrieve information from the Optim Directory. These utilities are described in “Informational Control Statements.”
- Maintain Optim objects in the Directory. These utilities are described in “Utilities to Maintain Optim Objects” on page 19.
- Manage Archive File entries and files. These utilities are described in “Archive Entry and File Management” on page 30.
- Migrate Optim objects. These utilities are described in “Utilities to Migrate Optim Objects” on page 45.

Note: An object or archive security exit enabled from the Site Options panel is called, when appropriate, during execution of the Batch Utility. If you use a security exit distributed with Optim, objects or Archive Files designated as PRIVATE cannot be accessed for any purpose and objects or Archive Files designated as READONLY cannot be modified or deleted using the Batch Utility.

Informational Control Statements

Use the following control statements to obtain information about Directory objects, Archive File and Archive Log entries, and Archive Files.

These statements and their functions are **DIRECTORY** and **REPORT**.

DIRECTORY

Use a DIRECTORY or DIR statement to generate a report of objects or entries in the Optim Directory.

Such a report can be useful to determine, for example, if any Column Maps are affected by changes to a database table. As further examples, you might obtain reports of

- Access Definitions modified by a particular user.
- Legacy Tables created by a specific user or associated with a specified data set.

You can generate summary or detailed reports about Access Definitions, Archive Collections, Column Maps, Table Maps, Legacy Tables, Optim or DB2 Primary Keys or Relationships, Compare Definitions, and extract or archive file entries. The report can be limited to one type of object or encompass all objects that match the criteria. The contents of the report are based on the parameters and operand values you specify. If you include multiple DIRECTORY statements in the JCL, the listing for each statement begins on a new page.

For clarity, parameters for archive files and archive log entries are described separately from parameters for Optim and certain DB2 objects. (For archive files and entries, see "DIRECTORY" on page 30.)

DIRECTORY Statement for Optim and Certain DB2 Objects

To generate a report of Optim or certain DB2 objects, use the DIRECTORY statement syntax as follows:

```
DIR(ECTORY)
  TYPE { ALL | AC | AD | CD | CM | ED | EXT | LT | PK | R | RD | TM | PR }
  NAME objectname
  [IGNORE_TYPE  { NO | YES }]
  [TABLE [ cid.] tblname]
  [COLUMN colname]
  [MODIFIEDBY modifiedby]
  [DESC desc]
  [DSNAME dsname]
  [ACTIONS { NO | YES }]
  [LEGACY_TABLE [ cid.] ltname]
  [FIELD fieldname]
  [DBDNAME dbdname]
  [SEGNAME segname]
  [ADNAME adname]
  [IMSID imsid]
  [PSBNAME psbname]
  [PNS { NO | YES }]
  [ARC { NO | YES }]
  [START_TABLE[ cid.]tblname]
  [DB2 { NO | YES }]
  [GENERIC { NO | YES }]
  [SORT ([NAME, A | D } ] [GROUP, { A | D } ] [DESC, { A | D }]
      [DATE, { A | D } ] [MODIFIEDBY, { A | D } ] )]
  [DETAIL{NO | YES}]
  [PRINTOPTS([ACT] [,ARC] [,COL] [,NONE] [,SEL] [,SQL])]
  [NEWPAGE{NO | YES}]
```

The parameters and values for operands are:

Required Criteria Parameters

TYPE Type of objects to list. This parameter must be included in the DIRECTORY statement with one of the following:

ALL List all object types that match the criteria.

Note: If criteria parameters are specified and IGNORE_TYPE is YES, only objects matching the criteria are listed.

AC List all Archive Collections that match the criteria.

AD List all Access Definitions that match the criteria.

CD List all Compare Definitions that match the criteria.

CM List all Column Maps that match the criteria.

ED List all Environment Definitions that match the criteria.

EXT List all extract files registered in the Directory that match the criteria.

Note: See the *Move User Manual* section on Import Extract Process for details.

LT List all Legacy Tables that match the criteria.

PK List all Primary Keys that match the criteria.

R List all Relationships that match the criteria.

RD List all Retrieval Definitions that match the criteria.

TM List all Table Maps that match the criteria.

PR List all Column Map Procedures that match the criteria.

NAME
Name identifying one or more objects. This parameter and the *objectname* operand must be included in the DIRECTORY statement.

objectname
Specify an explicit value or a pattern, using DB2 LIKE syntax. If TYPE is ALL, the maximum number of levels in the object name is two.

Optional Processing Keyword

IGNORE_TYPE
Indicate whether object types for which the specified criteria parameters do not apply should be included in the report. (Use IGNORE_TYPE only when TYPE is ALL.)

NO Include object types that match the specified criteria parameters in addition to all other object types.

YES Include only object types that match the specified criteria parameters. (Default)

Optional Criteria Parameters:
Use the following parameters to further refine criteria for objects to be listed.

– For DB2 and Optim Objects

TABLE
Name of table referenced in objects to be listed.

*cid***.***tblname*
The table name. Prefix with Creator ID if needed to identify the table. Specify as an explicit value or pattern, ending with %.

COLUMN
Name of a column referenced by objects other than Compare Definitions or in a table referenced by the objects to be listed.

colname
The column name.

– For All Optim Objects

MODIFIEDBY
TSO ID of the last person to modify the objects to be listed.

modifiedby
Specify as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the objects to be listed.

desc Specify as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

– For ADs, Compare Definitions, Legacy Tables, Table Maps, Environment Definitions, or Retrieval Definitions

DSNAME
Data set name referenced in listed objects. This parameter applies if TYPE is AD, CD, LT, TM, ED, or RD.

The referenced data set may be a Point-and-Shoot File, a Compare File, an Extract File, or an Archive File. For Legacy objects, the referenced data set may be a data file that is the data source for a Legacy Table, an IBM IMS™ program library, or

DBD/PSB (in the case of an Environment Definition), or an IMS database data set name (in the case of a Retrieval Definition).

Note: For a Retrieval Definition, if you are using dynamic allocation, specify the IMS database data set name as '$MDA' or '$OMITTED' if a data set name is not specified.

dsname
Specify the fully qualified data set name, without quotes, as an explicit value or a pattern, using DB2 LIKE syntax.

– For Access Definitions or Table Maps

ACTIONS
Actions attributes for listed Access Definitions or Table Maps. Omit the ACTIONS parameter to list Access Definitions or Table Maps without regard to actions attributes.

NO List Access Definitions or Table Maps that do not include actions.

YES List Access Definitions or Table Maps that include actions.

LEGACY_TABLE
Name of a Legacy Table referenced in Access Definitions or Table Maps to be listed.

*cid***.***ltname*
The Legacy Table name. Prefix as needed to identify the Legacy Table. Specify as an explicit value or a pattern, ending with %.

– For Legacy Tables

FIELD Name of the field in a Legacy Table referenced by the objects to be listed.

fieldname
The field name.

DBDNAME
Name of the DBD associated with a Legacy Table to be listed.

dbdname
Specify as an explicit value (i.e., you cannot specify a pattern).

SEGNAME
Name of the segment associated with a Legacy Table to be listed.

segname
Specify as an explicit value (i.e., you cannot specify a pattern).

– For Table Maps

ADNAME
List Table Maps created on the basis of an Access Definition.

adname
The name of the Access Definition. Specify as an explicit value or a pattern ending with %.

– For IMS Environment Definitions

IMSID
Name of the IMS System ID referenced in Environment Definitions to be listed.

imsid The IMS System ID.

– For IMS Retrieval Definitions

PSBNAME
Name of the PSB referenced in Retrieval Definitions to be listed.

psbname
The name of the PSB.

– For Access Definitions Only

PNS Indicate whether to list Access Definitions that use the Point-and-Shoot Start Table option. Use PNS only if TYPE is AD. Omit the PNS parameter to list Access Definitions without regard to the Point-and-Shoot Start Table option.

NO List only Access Definitions that do not use the Point-and-Shoot Start Table option.

YES List only Access Definitions that use the Point-and-Shoot Start Table option.

ARC Indicate whether to list Access Definitions with Archive criteria. Use this parameter only if TYPE is AD. Omit the ARC parameter to list Access Definitions without regard to archive criteria.

NO List Access Definitions without archive criteria.

YES List only Access Definitions for which archive criteria are specified.

START_TABLE
Name of the Start Table associated with the Access Definition to be listed.

cid.tblname
Creator ID and Start Table. Specify as an explicit value or a pattern ending with %.

– For Primary Keys or Relationships

DB2 Type of primary keys or relationships to list. Use this parameter only if TYPE is PK or R.

NO List only Optim primary keys or relationships.

YES List Optim and DB2 primary keys or relationships.

If you omit the DB2 parameter, only Optim primary keys or relationships are listed.

– For Access Definitions, Primary Keys or Relationships

GENERIC
List generic objects or objects that reference generic objects. Use this parameter if TYPE is AD, PK, or R. Omit the GENERIC parameter to list Access Definitions, primary keys, or relationships without regard to generic attributes or references.

NO List only objects without generic attributes or references.

YES List only generic primary keys, generic relationships, or Access Definitions that reference generic relationships.

Report Formatting Keywords

Use the following optional parameters to format the Directory Report.

SORT The sequence for list of objects in the report. If you do not specify a SORT parameter, the items are listed in ascending order, by name.

NAME
In order by name.

GROUP
In order by group.

DESC In order by description.

DATE In order by date modified.

MODIFIEDBY
In order by TSO ID of the user that last modified the object.

For all of the SORT parameters, valid values are:

A In ascending order (default).

D In descending order.

– For Detailed List

DETAIL
The level of detail in the report.

NO Report includes only Name, Description, Owner, Last Modified Date, and (if activated for site) Security Status for each listed object. This is also the information provided if you do not use the DETAIL parameter.

YES Report includes Name, Description, Owner, Creation or Last Modified Date, and Security Status for each listed object. The report also includes the following information, according to the type of object listed:

AC Archive Files in each Archive Collection

AD

- Default Creator ID
- Start Table name
- Optional information (see PRINTOPTS)
- Table List
- Relationship Usage information

CD

- Source 1 Tables
- Source 2 Tables
- Source 1 Access Definition, if any
- Source 2 Access Definition, if any
- Source 1 Extract File
- Source 2 Extract File

CM

- Source File name
- Source Table name with column names and data types
- Destination Table name with column names and data types
- Status of mapping for each column

PK(OPT)

- Table Name
- Key Column Names
- Key Column Attributes

PK(DB2)

- Table Name
- Key Column Names
- Key Column Attributes

PR

- Procedure Definition
- Description
- Modified By
- Last Modified
- Security Status
- Procedure CCSID
- Runtime CCSID
- print() Destination

R(OPT)

- Parent Table name with column names and data types
- Child Table name with column names and data types

R(DB2)

- Constraint Name
- Parent Table name
- Child Table name

TM

- Source Creator ID
- Source Table names
- Source Input Type
- Source Access Definition name
- Extract File DSN
- Destination Creator ID
- Destination Table names
- Column Map Creator ID
- Column Map names
- Action text (Optional, see PRINTOPTS)
- Validation rule

PRINTOPTS

Report includes detailed list of Access Definitions or Table Maps. Use this parameter if TYPE is AD or TM and DETAIL is YES.

- For Access Definitions

COL List all information for each Access Definition.

NONE

List all information for each Access Definition except column information (default).

SEL List all information for each Access Definition, but limit column information to columns with selection criteria.

SQL List all information for each Access Definition, but limit column information to columns with SQL.

ARC List all information for each Access Definition, but limit column information to columns with archive criteria.

- For Access Definitions and Table Maps

ACT List all information for each Action.

If you do not use the PRINTOPTS parameter, detailed listings of Access Definitions have no column information. Multiple values must be enclosed in parentheses and separated by commas.

NEWPAGE
Page break processing for the report. Use this parameter if DETAIL is YES.

NO Begin a new page on the basis of the PAGESIZE specification or default only. See PAGESIZE for further information.

YES In addition to normal page breaks, begin a new page for each object in the list.

Examples

The following are examples of DIRECTORY statement usage:

1. To create a listing of all Table Maps in the Directory, specify:

   ```
   DIR TYPE TM  NAME  %
   ```

2. To create a listing of Access Definitions with names prefixed by “FOPDEMO” that use the Point-and-Shoot Start Table option and include information about tables referenced in each Access Definition, specify:

   ```
   DIR TYPE AD NAME FOPDEMO.%  PNS  YES  DETAIL YES
   ```

3. To create a listing of all Access Definitions that refer to the table CUSTOMERS, specify:

   ```
   DIRECTORY TYPE  AD  NAME  %
      TABLE  CUSTOMERS
   ```

4. To create a listing of all objects that refer to the column CUSTNAME or refer to tables that include the column CUSTNAME, specify:

   ```
   DIRECTORY TYPE  ALL  NAME  %
      COLUMN CUSTNAME
   ```

5. To create a listing of all extract files registered in the Directory, specify:

   ```
   DIRECTORY TYPE EXT NAME %
   ```

REPORT

Use a REPORT statement to generate a report about the contents of a specific Archive, Control, or Extract File.

A report is useful to confirm that the proper data is deleted after archiving or to obtain summary information about a file. You can also obtain information about the Access Definition used to select data for an Archive or Extract File.

```
REP(ORT)
  NAME explicitfilename
  RPTTYPE { DATA | SUM } [ AD | ARC | DEL | EXT ]
  [ DDNAME ddname ]
  [ TBLNAME ( cid.tblname1, cid.tblname2, . . . ) ]
  [ EXCLUDE_TBLNAME (cid.tblname1, cid.tblname2, . . . ) ]
  [ PRINTOPTS ( [ ACT ] [, ARC ] [, COL ] [, NONE ] [, SEL ] [, SQL ] ) ]
  [ ERROR_ROWS_ONLY { NO | YES } ]
  [ ROW_STATUS { NO | YES | EXPlain } ]
  [ INSERT_STATUS { NO | YES } ]
  [ IMS_KEY { NO | YES | n } ]
```

```
[ WRAP_LINE { NO | YES } ]
[ TABLE_COUNTS { NO | YES } ]
[ VAR_CHAR_DELIMITER c ]
[ NULL_CHARACTER c ]
```

The parameters and operands in the REPORT statement are used to identify the Archive, Extract, or Control File and format the report. If the parameter specified is inappropriate for the identified file, an error occurs. For example, if RPTTYPE is AD for a Control File, or if RPTTYPE is DEL for an Extract File, an error will occur.

Note:

- You can create a REPORT for an extract file if the file is registered in the Directory and contains data rows.
- An extract file on tape can only be browsed by creating a REPORT.

Required Criteria Parameters

NAME

Name of the Archive, Extract, or Control File for which the report is generated. This parameter and the *explicitfilename* operand must be included in the REPORT statement.

explicitfilename

Specify the fully qualified name of a valid Archive, Extract, or Control File.

RPTTYPE

Contents of the report. This parameter must be included in the REPORT statement with one of the following:

DATA Data from the file.

SUM Only table names and row counts for data in the file.

Also specify one or more of the following:

ARC Archived data. Use only if NAME refers to an Archive File.

DEL Archived data for which delete after archive is specified. Use only if NAME refers to an Archive File.

AD Only the Access Definition used to select data in the file. Use AD only if NAME refers to an archive file or to an extract file registered in the Directory.

EXT Extract file. Use EXT only if NAME refers to an extract file registered in the Directory.

Optional Criteria Parameters

Use the following parameters to format the report for the file.

All Files

DDNAME

DDNAME for the Report data set. (The default DDNAME for a Report file, used if you do not provide a DDNAME parameter, is PSDFRPRT.)

ddname

Specify the DDNAME for the Report file.

Archive or Extract Files

TBLNAME

Name of a table or tables included in the report. Omit this parameter to create a report on data from all tables in the file. TBLNAME does not apply when

RPTTYPE is SUM since the table names and row counts are displayed with this report type. (You cannot use TBLNAME and EXCLUDE_TBLNAME in the same statement.)

cid.tblname
The fully qualified name of one or more tables in the file. Specify as an explicit value or a pattern, ending with %.

EXCLUDE_TBLNAME
Name of table or tables to be omitted from the report. All other tables in the Archive or Extract file are included. You cannot use EXCLUDE_TBLNAME and TBLNAME in the same statement.

cid.tblname
The fully qualified name of one or more tables in the file. Specify as an explicit value or a pattern, ending with %.

PRINTOPTS
Report includes detailed information about the Access Definition used to select data in the file. Use this parameter if RPTTYPE is AD.

COL List all information for the Access Definition.

NONE
List all information, except column information.

SEL List all information, but limit column information to columns with selection criteria.

SQL List all information, but limit column information to columns with SQL.

ARC List all information, but limit column information to columns with Archive criteria.

ACT List all information for each Action in the Access Definition.

If you do not use the PRINTOPTS parameter, the report has no column information. Multiple values must be enclosed in parentheses and separated by commas.

Control Files

ERROR_ROWS_ONLY
Display of error rows in the Control File.

NO Include all rows (default).

YES Include only rows with errors.

ROW_STATUS
Display row status from the Control File.

NO Do not include row status (default).

YES Include row status.

EXPlain
Print explanation of row status.

INSERT_STATUS
Display insert status from the Control File.

NO Do not include insert status (default).

YES Include insert status.

IMS Tables

IMS_KEY
Display of IMS concatenated key for IMS tables only. (This parameter is ignored for non-IMS tables.)

NO Do not display the IMS key (default).

YES Display the entire IMS key.

n Display the first *n* characters of the IMS key. (Display the entire key if *n* exceeds the key length.)

All Files

WRAP_LINE
Handling if rows in the file are longer than the output lines.

NO Change output file characteristics to accommodate the longest record in the file. (If you specify this option and direct the report to a disk or tape data set, the data set must not have an LRECL value specified.)

YES Wrap lines on the basis of the record length of the output file (default).

TABLE_COUNTS
Display table counts in the report. A table count is included unless otherwise indicated.

NO Do not include table counts in the report.

YES Include table counts in report.

VAR_CHAR_DELIMITER
The character used to delimit variable character data, specified as:

c ; : ? ! & % # @

NULL_CHARACTER
The character used to indicate a column is NULL, specified as:

c ? ! & % # @

Examples

The following are examples of the REPORT statement:

1. To produce a report for an Extract File showing the Access Definition and selection criteria used to create it, specify:

```
REPORT NAME  PSTUSER.OUTSTOCK  RPTTYPE  AD
    PRINTOPTS  SEL
```

2. To produce a report for a Control File showing only error rows and explanations of their status, specify:

```
REPORT NAME  PSTUSER.NEWCST21  RPTTYPE  DATA
   ERROR_ROWS_ONLY  YES   ROW_STATUS  EXP
```

Utilities to Maintain Optim Objects

The Directory contains Optim objects such as Access Definitions, Archive Collections, Table Maps, Column Maps, Legacy Tables, Optim Primary Keys and Relationships, Compare Definitions, Environment Definitions, and Retrieval Definitions. For example, you can use a Batch Utility control statement to copy Access Definitions modified by a specific user in order to create a set of identical Access Definitions with different names. Other Batch Utility control statements allow you to delete or rename Optim objects.

The Directory also contains Archive File entries that make it possible to search, restore, or browse data in Archive Files and Archive log entries that record archiving activity. Batch Utility control statements allow you to create or delete Archive File entries or change the index information for an entry. You can also delete Archive log entries. Control statements that can be used to maintain objects and Archive entries in the Directory and the functions of the control statements are:

"DELETE"
: Delete Optim objects, Archive File entries, or Archive log entries from the Directory. (When deleting Archive File entries, you can optionally delete the related z/OS data sets.)

"COPY" on page 24
: Copy Optim objects in the Directory, name the new objects, and generate a report about the copied objects.

"RENAME" on page 25
: Change the names of Optim objects in the Directory and generate a report about the renamed objects.

"GENERATE" on page 27
: Define and update Legacy Tables in batch.

DELETE

Use a DELETE statement to remove one or more of these objects from the Directory: Access Definitions, Archive Collections, Column Maps, Table Maps, Legacy Tables, Primary Keys, Relationships, Compare Definitions, Environment Definitions, Retrieval Definitions, entries for extract or archive files. For example, Optim objects exported to a Directory in a different subsystem can be deleted from the original Directory.

The Optim objects or entries to be removed are selected according to keywords and values specified as operands. The keywords and operands used to delete archive file or log entries are described separately from those for Optim objects. See "DELETE" on page 36 for a discussion of the use of the DELETE statement for archive entries.

DELETE Statement for Optim Objects

To delete Optim objects, use the DELETE statement syntax as follows:

```
DELETE
  TYPE { AC | AD | CD | CM | ED | EXT | LT | PK | PR | R | RD | TM  }
  NAME objectname [ TABLE [cid. ] tblname]
  [COLUMN colname]
  [MODIFIEDBY modifiedby]
  [DESC desc]
  [DSNAME dsname]
  [ACTIONS { NO | YES }]
  [LEGACY_TABLE [cid.]ltname]
  [FIELD fieldname]
  [DBDNAME dbdname]
  [SEGNAME segname]
  [ADNAME adname]
  [IMSID imsid]
  [PSBNAME psbname]
  [PNS { NO | YES }]
  [ARC { NO | YES }]
  [START_TABLE [cid.]tblname]
  [DB2 { NO | YES }]
  [GENERIC { NO | YES }]
```

Except for TYPE ALL and IGNORE_TYPE, which are not valid for a DELETE statement, the keywords and operands are identical to those in the DIRECTORY statement. You can use these keywords and operands in a DIRECTORY statement to list and review candidates for deletion before submitting a job with the same keywords and operands in a DELETE statement.

Required Criteria Keywords

TYPE Type of Optim objects to be deleted. This keyword must be included in the DELETE statement with one of the following:

AC Delete all Archive Collections that match the criteria.

AD Delete all Access Definitions that match the criteria.

CD Delete all Compare Definitions that match the criteria.

CM Delete all Column Maps that match the criteria.

ED Delete all Environment Definitions that match the criteria.

EXT Delete all extract file entries that are registered in the Directory and match the criteria.

LT Delete all Legacy Tables that match the criteria.

PK Delete all Optim Primary Keys that match the criteria.

PR Delete all Column Map Procedures that match the criteria.

R Delete all Optim relationships that match the criteria.

RD Delete all Retrieval Definitions that match the criteria.

TM Delete all Table Maps that match the criteria.

NAME

Name identifying one or more Optim objects. This keyword and the *objectname* operand must be included in the DELETE statement.

objectname

Specify as an explicit value or a pattern, using DB2 LIKE syntax.

Optional Criteria Keywords

Use the following keywords to further refine the criteria for objects to be deleted.

For All Optim Objects

TABLE

Name of a table referenced in objects to be deleted.

*cid***.***tblname*

The table name. Prefix with Creator ID, if needed, to identify the table. Specify as an explicit value or a pattern ending with %.

COLUMN

Name of the column referenced by the objects or in a table referenced by the objects to be deleted.

colname

The column name.

MODIFIEDBY

TSO ID of the last person to modify the Optim objects. Use this keyword only if TYPE is AD, CM, TM, R, PK, or CD.

modifiedby

Specify as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the Optim objects to be deleted.

desc Specify as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

For Compare Definitions, Legacy Tables, ADs, Table Maps, Environment Definitions, or Retrieval Definitions

DSNAME

Data set name referenced in listed objects. This keyword applies if TYPE is AD, CD, LT, TM, ED, or RD.

The referenced data set may be a Point-and-Shoot File, a Compare File, an Extract File, or an Archive File. For Legacy objects, the referenced data set may be a data file that is the data source for a Legacy Table, an IMS program library, or DBD/PSB (in the case of an Environment Definition), or an IMS database data set name (in the case of a Retrieval Definition).

Note: For a Retrieval Definition, if you are using dynamic allocation, specify the IMS database data set name as '$MDA' or '$OMITTED' if a data set name is not specified.

dsname

Specify the fully qualified data set name, without quotes, as an explicit value or a pattern, ending with %.

For Access Definitions or Table Maps

ACTIONS

Actions attributes for deleted Access Definitions or Table Maps. Omit the ACTIONS keyword to delete Access Definitions or Table Maps without regard to actions attributes.

NO Delete Access Definitions or Table Maps that do not include actions.

YES Delete Access Definitions or Table Maps that include actions.

LEGACY_TABLE

Name of a Legacy Table referenced in Access Definitions or Table Maps to be deleted.

*cid***.***ltname*

The Legacy Table name. Prefix as needed to identify the Legacy Table. Specify as an explicit value or a pattern, ending with %.

For Legacy Tables

FIELD Name of a field in a Legacy Table referenced by the objects to be deleted.

fieldname

The field name.

DBDNAME

Name of the DBD associated with a Legacy Table to be deleted.

dbdname

Specify the DBD name as an explicit value (i.e., you cannot use wildcards).

SEGNAME

Name of the segment associated with a Legacy Table to be deleted.

segname

Specify the segment name as an explicit value (i.e., you cannot use wildcards).

For Table Maps

ADNAME

Delete Table Maps created on the basis of an Access Definition.

adname

The name of the Access Definition. Specify as an explicit value or a pattern, ending with %.

For IMS Environment Definitions

IMSID

Name of the IMS System ID referenced in an Environment Definition to be deleted.

imsid The IMS System ID.

For IMS Retrieval Definition

PSBNAME
Name of the PSB referenced in a Retrieval Definition to be deleted.

psbname
The name of the PSB.

For Access Definitions

PNS Indicate whether to delete Access Definitions that use the Point-and-Shoot Start Table option. Use PNS only if TYPE is AD. Omit the PNS keyword to delete Access Definitions without regard to the Point-and-Shoot Start Table option.

NO Delete only Access Definitions that do not use the Point-and-Shoot Start Table option.

YES Delete only Access Definitions that use the Point-and-Shoot Start Table option.

ARC Indicate whether to delete Access Definitions with Archive criteria. Use this keyword only if TYPE is AD. Omit the ARC keyword to delete Access Definitions without regard to archive criteria.

NO Delete Access Definitions without regard to archive criteria.

YES Delete only Access Definitions for which archive criteria are specified.

START_TABLE
Name of the Start Table in the Access Definition to be deleted.

cid.tblname
Name of the Creator ID and Start Table. Specify as an explicit value or a pattern ending with %.

DB2 Type of primary keys or relationships to be deleted. Use this keyword only if TYPE is PK or R.

NO Delete only Optim primary keys or relationships.

YES Delete Optim and DB2 primary keys or relationships.

If you do not use the DB2 keyword, only Optim primary keys or relationships are deleted.

For Access Definitions, Primary Keys or Relationships

GENERIC
Delete generic objects or objects that reference generic objects. Use this keyword if TYPE is AD, PK, or R. Omit the GENERIC keyword to delete Access Definitions, primary keys, or relationships without regard to generic attributes or references.

NO Delete only objects without generic attributes or references.

YES Delete only generic primary keys, generic relationships, or Access Definitions that reference generic relationships.

Examples

To delete Column Maps with names that are prefixed with FOPDEMO, specify:

```
DELETE  TYPE CM  NAME  FOPDEMO.%
```

To delete an entry from the Directory for a specific extract file, specify:

```
DELETE TYPE EXT NAME PSTJS.BATCH.XF
```

COPY

Use a COPY statement to create one or more Optim objects that are exactly like existing objects that match the criteria, but have different names.

This statement also generates a report of the new objects in the Directory listing file.

```
COPY
  TYPE { ALL | AC | AD | CD | CM | ED | LT | PK | PR | R | RD | TM }
  FROM objectname
  TO objectname
  [ MODIFIEDBY modifiedby ]
  [ DESC desc ]
  [ REPLACE { NO | YES } ]
```

Required Criteria Keywords

The keywords and operands in the COPY statement are:

TYPE Type of objects to be copied. This keyword must be included in the COPY statement with one of the following:

ALL Copy all objects.

AC Copy all Archive Collections that match criteria.

AD Copy all Access Definitions that match criteria.

CD Copy all Compare Definitions that match criteria.

CM Copy all Column Maps that match criteria.

ED Copy all Environment Definitions that match the criteria.

LT Copy all Legacy Tables that match criteria.

PK Copy all Optim Primary Keys that match criteria.

PR Copy all Column Map Procedures that match criteria.

R Copy all Optim relationships that match criteria.

RD Copy all Retrieval Definitions that match the criteria.

TM Copy all Table Maps that match criteria.

FROM

Name identifying source objects to be copied. This keyword and *objectname* must be included in the COPY statement.

objectname

Specify as an explicit value or a pattern, using DB2 LIKE syntax. An ERROR condition occurs if a matching object is not in the Directory. If TYPE is ALL, the maximum number of levels allowed in *objectname* is two.

TO Name identifying new objects. This keyword and *objectname* must be included in the COPY statement.

objectname

Specify as an explicit value or a pattern, using DB2 LIKE syntax. If TYPE is ALL, the maximum number of levels allowed in *objectname* is two. (You can change only the first two parts of an Access Definition name.)

Optional Criteria Keywords

MODIFIEDBY

TSO ID of the last person to modify the source objects.

modifiedby
Specify as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive description of the Optim objects.

desc Specify as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

Optional Replace Keyword

REPLACE
Action taken when the name of a new object (taken from the *objectname* operand for the TO keyword) matches the name of an existing object in the Directory.

NO Do not replace.

YES Replicate the object and replace any existing object of the same type with the same name.

If you do not use the REPLACE keyword, existing objects are retained.

Examples

The following are examples of the COPY statement usage:

1. To replicate all Column Maps with names prefixed with “FOPDEMO” and name the new Column Maps using the prefix “FOPPROD,” replacing any existing Column Maps that have matching names, specify:
```
COPY TYPE  CM  FROM  FOPDEMO.%  TO
   FOPPROD.%  REPLACE  YES
```
2. To replicate all Access Definitions with names prefixed with “FOPDEMO” and created or last modified by WMD, specify:
```
COPY TYPE  AD  FROM  FOPDEMO.%.%  TO
   FOPPROD.%.%  MODIFIEDBY  WMD
```

RENAME

Use a RENAME statement to change the name of one or more Optim objects and generate a report of the renamed objects in the Directory listing file.

Security requirements are the same as those for the online Rename function.

```
RENAME
  TYPE { ALL | AC | AD | CD | CM | ED | LT | PK | PR | R | RD | TM }
  FROM objectname
  TO objectname
  [ MODIFIEDBY modifiedby ]
  [ DESC desc ]
  [ REPLACE { NO | YES } ]
```

Required Criteria Keywords
The keywords and operands in the RENAME statement are:

TYPE Type of objects to rename. This keyword must be included in the RENAME statement with one of the following:

ALL Rename all objects.

AC Rename all Archive Collections that match criteria.

AD Rename all Access Definitions that match criteria.

CD Rename all Compare Definitions that match criteria.

CM Rename all Column Maps that match criteria.

ED Rename all Environment Definitions that match the criteria.

LT Rename all the Optim Legacy Tables that match criteria.

PK Rename all Optim Primary Keys that match criteria.

PR rename all Column Map procedures that match criteria.

R Rename all Optim relationships that match criteria.

RD Rename all Retrieval Definitions that match the criteria.

TM Rename all Table Maps that match criteria.

FROM

Name identifying source objects to rename. This keyword and the *objectname* operand must be included in the RENAME statement.

objectname

Specify as an explicit value or a pattern, using DB2 LIKE syntax. If TYPE is ALL, the maximum number of levels allowed in *objectname* is two.

TO Name identifying objects after renaming. This keyword and the *objectname* keyword must be included in the RENAME statement.

objectname

Specify as an explicit value or a pattern, using DB2 LIKE syntax. If TYPE is ALL, the maximum number of levels allowed in *objectname* is two.

Optional Criteria Keywords

MODIFIEDBY

TSO ID of the last person to modify the Optim objects.

modifiedby

Specify as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive description of the Optim objects.

desc Specify as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

Optional Replace Keyword

REPLACE

Action taken when the name of a new object (taken from the *objectname* operand for the TO keyword) matches the name of an existing object in the Directory.

NO Do not rename.

YES Rename the source object and replace the similarly named existing object.

If you do not use the REPLACE keyword, existing objects are retained.

Examples

The following are examples of the RENAME statement usage:

1. To rename all Column Maps with names prefixed with “PSTDEMO” and replace any existing Column Maps with matching names, specify:

```
RENAME  TYPE  CM  FROM  PSTDEMO.%  TO
   PSTPROD.%  REPLACE  YES
```

2. To rename all Access Definitions with names prefixed with “PSTDEMO” and created or last modified by WMD, specify:

```
RENAME  TYPE  AD  FROM  PSTDEMO.%.%  TO
   PSTPROD.%.%  MODIFIEDBY  WMD
```

GENERATE

Use a GENERATE statement to define and update Legacy Tables.

You cannot override or change the field names in the Legacy Table using the GENERATE batch utility. The Legacy Table field names are the same as the copybook field names unless they are truncated because of length restrictions.

Note: To get the GENERATE statement report, you must allocate DDNAME PSDFADIR in your batch job.

```
GENERATE
  TYPE LT
  TABLE cid.ltname
  [ DESC //description// ]
  [ SECURITY { PUBLIC | READONLY | PRIVATE ]
  [ CRIT //recordcriteria// ]
  [ LEGDSN dsname ]
  LANGUAGE { P | C }
  LEGTYPE { D | I }
      DBDNAME dbdname
      SEGNAME segname
  COPYBOOK dsname(mem)
  [ LEGEXIT ioexitname ]
  [ SBCS_CCSID ccsid ]
  [ MBCS_CCSID ccsid ]
  [ DBCS_CCSID ccsid ]
  [ GENERATE_ERROR { STOP | CONT } ]
  [ REPLACE { NO | YES } ]
  [ DETECT_ISO { NO | YES } ]
```

Required Criteria Keywords

The keywords and values for operands are:

TYPE TYPE must be included in the GENERATE statement with the following operand, LT.

TABLE

Legacy Table name as an explicit value.

cid.ltname

Prefix with Creator ID if necessary to identify the Legacy Table. If Creator ID is omitted, the batch job Creator ID is used.

LANGUAGE

The language of the source copybook.

P PL/I

C COBOL

LEGTYPE

The type of the Legacy source data set.

D A sequential or VSAM data set

I An IMS file

DBDNAME

The DBD name for the IMS database with which the Legacy Table is associated. The DBD must be in a DBD Library referenced in the Environment Definition that matches the creator ID of the Legacy Table. (Required if LEGTYPE is I.)

SEGNAME

The name of the segment in the specified DBD with which the Legacy Table is associated. (Required if LEGTYPE is I.)

COPYBOOK

The name of the copybook data set/member name.

*dsname***(***mem***)**

Each copybook member must have only one Level 01 field. In this way, each member results in one Legacy Table definition. Field names that are more than 18 bytes long are truncated to 18 bytes. You cannot override or change the field names in the Legacy Table. They are the same as the copybook field names unless they are truncated.

Optional Criteria Keywords

Use the following keywords as necessary to further refine the criteria for Legacy Table generation.

DESC Description of the Legacy Table. Enter 1 - 40 characters enclosed in double slashes.

SECURITY

Level of security assigned to the Legacy Table.

PUBLIC

All can use and modify (default)

READONLY

All can use but only owner can modify

PRIVATE

Only owner can use and modify

CRIT The criteria used to match the Legacy Table with a record in the source data set. (Included only when specified in the Legacy Table definition.)

LEGDSN

The fully qualified name of the Legacy source data set.

Note: When an IMS database data set is the source data set for this Legacy Table and you are using dynamic allocation, specify '$MDA' as the data set name. You can omit the `LEGDSN` parameter if the site option `Require IMS data set name` is set to `N`.

LEGEXIT

The name of the I/O Exit Load module, if different from FOPIMIOX or FOP2DIOX or the default exit name specified in the Site Options.

SBCS_CCSID

Single-byte Coded Character Set Identifier (CCSID) for the Legacy Table

MBCS_CCSID

Multi-byte CCSID for the Legacy Table

DBCS_CCSID

Double-byte CCSID for the Legacy Table

REPLACE

Replace existing Legacy Table definition.

NO Do not replace existing definition (default).

YES Replace existing definition.

GENERATE_ERROR

Specifies processing when an error condition occurs during Legacy Table generation.

STOP Processing stops when an error condition occurs and further GENERATE operations are suppressed (default).

CONT

Processing continues when an error condition occurs and further GENERATE operations are executed.

DETECT_ISO

Specifies whether Optim looks for the $$FOPDAT, $$FOPTIM, and $$FOPSTP keywords when parsing copybooks to build Legacy Table definitions.

NO Optim does not look for the $$FOP*xxx* keywords when parsing copybooks (default).

YES Optim looks for the $$FOP*xxx* keywords when parsing copybooks. The $$FOPDAT, $$FOPTIM, and $$FOPSTP keywords can be included within copybook comments. The comment must appear on the line directly before the related field definition. The $$FOP*xxx* keyword must be separated by spaces from other text in the comment line.

The $$FOP*xxx* keywords indicate that the field following the comment is in ISO DATE, TIME, or TIMESTAMP format. When Optim finds the $$FOP*xxx* keyword, it sets the following field to the appropriate DAT, TIM, or STP type in the Legacy Table definition. For COBOL, the field must be defined as alphanumeric. For PL/I, the field must be defined as character. The field must also adhere to the following rules.

$$FOPDAT - ISO DATE: The field must have a length of 10 and be defined in YYYY-MM-DD format.

$$FOPTIM - ISO TIME: The field must have a length of 8 and be defined in HH:MM:SS format.

$$FOPSTP - ISO TIMESTAMP: The field must have a length of 26 and be defined in YYYY-MM-DD-HH.MM.SS.NNNNNN format.

The following examples illustrate the $$FOP*xxx* keyword comments and related field definitions.

COBOL copybook

```
 01 SAMPREC.
     02 CUSTNAME          PIC X(80).
*  $$FOPDAT
     02 SAMPLE-DATE       PIC X(10).
*  $$FOPTIM
     02 SAMPLE-TIME       PIC X(8).
*  $$FOPSTP
     02 SAMPLE-TIMESTAMP  PIC X(26).
```

PL/I copybook

```
1 SAMPREC,
   2 CUSTNAME           CHAR(80),
/* $$FOPDAT                                    */
   2 SAMPLE_DATE        CHAR(10),
/* $$FOPTIM                                    */
   2 SAMPLE_TIME        CHAR(8),
/* $$FOPSTP                                    */
   2 SAMPLE_TIMESTAMP   CHAR(26);
```

Example

Specify the following to generate a Legacy Table (PSTDEMO.SALES) using copybook/member, PSTQA.RT.COPYLIB(SALES).

Note: The following example illustrates an IMS file.

```
GENERATE
   TYPE LT
   TABLE PSTDEMO.SALES
   SECURITY PUBLIC
   LANGUAGE P
   LEGTYPE I
   DBDNAME SALEHIDM
   SEGNAME SALES
   COPYBOOK PSTQA.RT.COPYLIB(SALES)
   REPLACE YES
```

Archive Entry and File Management

Archive utilities allow you to manage Archive Files, Archive File entries, and Archive log entries in batch. These utilities also automate certain online functions such as those for reports of Archive Files and maintenance of Archive File or log entries and indexes.

These batch utilities help you maintain the Archive environment and are an efficient means of monitoring the Archive functions and operations in your environment. For example, if a column is added to a table in your database, you can use a batch utility to generate a report of Archive Files that include data from the table. The report can be a useful tool to help you decide how to define and name a Table Map for restoring data from those Archive Files. You can also generate a report of Archive log entries to monitor the frequency and number of Archive, Restore, Search, Browse, and other processes. This information can be useful in troubleshooting or fine tuning your archiving strategy.

The Archive batch processing statements are:

"DIRECTORY"
List Archive File entries or log entries in the Directory.

"DELETE" on page 36
Delete Archive File or log entries from the Directory. (When deleting Archive File entries, you can optionally delete the related z/OS data sets.)

"ARCHIVE_IMPORT" on page 41
Create an entry in the Directory for an imported Archive File. (An entry enables Archive functions, such as Browse or Search, for the corresponding Archive File.)

"ARCHIVE_UPDATE" on page 43
Add or delete index entries for an Archive File.

"OUTDD" on page 44
Indicates the DDNAME for the Directory listing data set.

DIRECTORY

To generate a list of Archive File or log entries in the Directory, use the DIRECTORY statement syntax as follows:

```
DIR(ECTORY)
  TYPE { ARC | ARCLOG }
  NAME filename
  [ IGNORE_TYPE { NO | YES } ]
  [ GROUP group ]
  [ DESC desc ]
  [ FROMDATE yyyy-mm-dd ]
  [ TODATE yyyy-mm-dd ]
  [ OLDER (n keyword [, n keyword,... ] ) ]
  [ FUNCTION ( [ ALL ] [, ARCHIVE ] [, SEARCH ] [, RESTORE ]
     [, BROWSE ] [, DELETE ] [, SUBSET ] [, LOAD ]
     [, CONVERT ] [, IMPORT ] [, UPDATE ] ) ]
  [ PROCESS_MODE { ONLINE | BATCH } ]
  [ PROCESSED_BY userid ]
  [ TABLE [ cid. ] tblname ]
[ COLUMN colname ]
[ ARCHIVEDBY userid ]
[ MISSINGFILE { NO | YES } ]
[ EXPIRE { ( n keyword [, n keyword, ... ] ) | explicitdate } ]
[ ADNAME { group.user.name | NONE } ]
[ SORT ( [ NAME, { A | D } ] [ GROUP, { A | D } ] [ DESC, { A | D } ]
   [ DATE, { A | D } ] [ ARCHIVEDBY, { A | D } ] ) ]
[ DETAIL { NO | YES } ]
[ PRINTAD { NO | YES } ]
[ PRINTOPTS ( [ ACT ] [, ARC ] [, COL ] [, NONE ] [, SEL ] [, SQL ] ) ]
[ NEWPAGE { NO | YES } ]
```

The keywords and values for operands are:

Required Criteria Keywords

TYPE Type of object to be listed. TYPE must be included in the DIRECTORY statement.

ARC List all Archive File entries that match other criteria.

ARCLOG
List Archive log entries that match other criteria.

NAME
Name identifying one or more Archive files or Archive log entries to be listed. This keyword is required for TYPE ARC and optional for TYPE ARCLOG.

filename
Specify as an explicit value or a pattern, using DB2 LIKE syntax.

Optional Processing Keyword

IGNORE_TYPE
Indicate whether object types for which the specified criteria keywords do not apply should be included in the report.

NO Include object types that match the specified criteria keywords in addition to all other object types.

YES Include only object types that match the specified criteria keywords (default).

If you do not use the IGNORE_TYPE keyword, the object is not included in the report.

Optional Criteria Keywords
Use the following keywords to further refine the criteria for entries to be listed.

For Archive File and Archive Log Entries

GROUP
Group designation for the entries to be listed.

group Specify as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the entries to be listed.

desc Specify as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

FROMDATE
The oldest date in a range of Archive File creation dates. Entries for selected Archive Files are listed.

yyyy-mm-dd
Specify in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use FROMDATE alone to select entries for all Archive Files created on or after that date.

- Use FROMDATE with TODATE to designate a range.

You can use FROMDATE or OLDER, but not both.

TODATE

The most recent date in a range of Archive File creation dates. Entries for selected Archive Files are listed.

yyyy-mm-dd

Specify in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use TODATE alone to select all entries for Archive Files created on or before that date.
- Use TODATE with FROMDATE to designate a range.

You can use TODATE or OLDER, but not both.

OLDER

An interval used to exclude entries from listing; that is, only entries for Archive Files that are older than the specified interval are listed. You can use OLDER, or FROMDATE and TODATE, but not both.

n The number of days, weeks, months, or years in the interval.

keyword

DAY(S)

WEEK(S)

MONTH(S)

YEAR(S)

You can use one or more keywords (DAYS, WEEK, etc.), each with an *n* operand, in any order.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

Examples:

- Specify OLDER (1 YEAR) on March 23, 1999, to list entries for Archive Files created on or before March 23, 1998.
- Specify OLDER (2 YEARS, 6 MONTHS) on June 30, 1999, to list entries for Archive Files created on or before December 31, 1996.

For Archive Log Entries Only

Use the following optional criteria keywords only when TYPE is ARCLOG. Using them when the TYPE is ARC causes an error condition unless IGNORE_TYPE is specified as YES.

FUNCTION

Process for which log entries are listed. If the FUNCTION keyword is not used, all log entries are listed. Enclose multiple function operands in parentheses and separate with commas.

ALL List all log entries.

ARCHIVE

List all Archive Process log entries.

SEARCH

List all Search Process log entries.

RESTORE

List all Restore Process log entries.

BROWSE
List all Browse Process log entries.

DELETE
List all Delete Process log entries.

SUBSET
List all Subset Process entries.

LOAD
List all Load Process entries.

CONVERT
List all Convert Process entries.

IMPORT
List all Import Process entries.

UPDATE
List all Update Process entries.

PROCESS_MODE
The mode in which processes were executed. If the PROCESS_MODE keyword is not used, log entries are listed without regard to mode.

ONLINE
List log entries for online processes.

BATCH
List log entries for batch processes.

PROCESSED_BY
TSO ID of the user that initiated the processes for which log entries are listed.

userid Specify any string that represents a user ID.

For Archive File Entries Only
Use the following optional criteria keywords only when TYPE is ARC. Using them when TYPE is ARCLOG causes an error condition unless IGNORE_TYPE is specified as YES.

TABLE
Name of a table included in Archive Files for which Archive File entries are listed.

cid.tblname
The table name. Prefix with Creator ID, if needed to identify the table. Specify as an explicit value or a pattern, ending with %.

COLUMN
Name of a column referenced by objects in Archive Files for which Archive File entries are listed.

colname
The column name. Specify as an explicit value or a pattern, using DB2 LIKE syntax.

ARCHIVEDBY
TSO ID of the user that created the Archive Files for which Archive File entries are listed.

userid Specify as an explicit value or a pattern, using DB2 LIKE syntax.

MISSINGFILE
File associations for listed Archive File entries. Omit the MISSINGFILE keyword to list Archive File entries without regard to the presence of a physical Archive File.

NO List Archive File entries without regard to the presence of an Archive File.

YES List only Archive File entries for which an Archive File is not present.

EXPIRE
Archive File expiration date for listed Archive File entries.

Note: You may specify multiple *n* keywords within parentheses. If you specify one *n keyword*, you do not need parentheses.

n The number of days, weeks, months, or years to be added to the Archive File creation date. Any Archive File entries for Archive Files with an earlier expiration date are listed.

keyword

DAY(S)

WEEK(S)

MONTH(S)

YEAR(S)

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

explicitdate
Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

ADNAME
Access Definition association for listed Archive File entries. Omit the ADNAME keyword to list Archive File entries without regard to the Access Definition.

group.user.name
List only Archive File entries for Archive Files created with the named Access Definition.

NONE
List only Archive File entries for Archive Files created with an unknown Access Definition. In other words, list Archive File entries for Archive Files created with version 4.1 or earlier of Archive for DB2.

Report Formatting Keywords

Use the following keywords to format the Directory Report.

SORT The sequence for the Archive File entries in the report. If you do not specify a SORT keyword, the entries are listed in ascending order, by name.

NAME
In order by name.

GROUP
In order by group.

DESC In order by description.

DATE In order by date created.

ARCHIVEDBY
In order by TSO ID of the user that created the Archive File. Use this operand only if TYPE is ARC.

For all of the SORT keywords, valid values are:

A In ascending order (default).

D In descending order.

Detailed List

DETAIL

The level of detail in the report. If you do not use the DETAIL keyword, only Name, Description, Owner, Creation Date, and Security Status for each Archive or log entry are listed in the report.

NO Report includes only Name, Description, Owner, Creation Date, and Security Status for each listed Archive entry.

YES Report includes Name, Description, Owner, Creation, and Security Status for each listed Archive entry. The report also includes tables in the corresponding Archive File, any index columns for each table, and low and high values for each column.

PRINTAD

Report includes information about the Access Definition used to select data in the Archive File.

NO Report does not include the Access Definition associated with the Archive File.

YES Report includes the Access Definition associated with the Archive File.

PRINTOPTS

Format for detailed information about the Access Definition associated with the Archive File. Use this keyword only if PRINTAD is YES.

- If you do not use the PRINTOPTS keyword, the report provides no column information for the Access Definition.
- Multiple values must be enclosed in parentheses and separated by commas.

COL List all information for the Access Definition.

NONE

List all information for the Access Definition except column information (default).

SEL List all information for the Access Definition, but limit column information to columns with selection criteria.

SQL List all information for the Access Definition, but limit column information to columns with SQL statement(s).

ARC List all information for the Access Definition, but limit column information to columns with archive criteria.

ACT List all information for each Action.

NEWPAGE

Page break processing for the report. Use this keyword only if DETAIL is YES.

NO Begin a new page on the basis of the PAGESIZE specification or default. (Default)

YES Begin a new page for each Archive entry in the list.

Examples

The following are examples of DIRECTORY statement usage:

1. To create a listing of all Archive File entries in the Directory, specify:

```
DIR TYPE  ARC  NAME  %
or
DIRECTORY  TYPE  ARC  NAME  %
```

2. To create a listing of all Archive File entries in the Directory and include Access Definition information, specify:

```
DIR  TYPE  ARC  NAME  %  PRINTAD  YES
```

3. To create a listing of all Archive File entries for files with names prefixed by “PSTARC” and include information about the tables and indexed columns for each listed entry, specify:

   ```
   DIR  TYPE  ARC  NAME  PSTARC.%  DETAIL YES
   ```

4. To create a listing of all Archive File entries for files created by the user named ACCOUNTS on November 1, 1998, or later, arranged in descending order by group and, within each group, in ascending order by name and date, specify:

   ```
   DIR  TYPE  ARC  NAME  %
      ARCHIVEDBY  ACCOUNTS
      FROMDATE  1998-11-01
      SORT (GROUP, D, NAME, DATE)
   ```

5. To create a listing of all Archive Log entries for batch processes initiated by the user PSTABC, specify:

   ```
   DIR  TYPE  ARCLOG
      FUNCTION  ALL
      PROCESS_MODE  BATCH
      PROCESSED_BY  PSTABC
   ```

6. To create a listing of Archive log entries for online Search Processes initiated by the user named PSTABC on December 1, 2000 or earlier, and arrange the listing in descending order by name and date, specify:

   ```
   DIR  TYPE  ARCLOG
      FUNCTION  SEARCH
      PROCESS_MODE  ONLINE
      PROCESSED_BY  PSTABC
      TODATE  2000-12-01
      SORT  (GROUP, D, NAME, DATE)
   ```

DELETE

Use a DELETE statement to delete Archive File or Archive log entries that satisfy criteria described by keywords and values for operands. To delete Archive File or log entries in the Directory, use the DELETE statement syntax as follows.

Note the optional keyword, FILE, used to delete any z/OS data sets that correspond to the deleted Archive File entries.

```
DELETE
  TYPE { ARC | ARCLOG }
  NAME filename
  [ GROUP group ]
  [ DESC desc ]
  [ FROMDATE yyyy-mm-dd ]
  [ TODATE yyyy-mm-dd ]
  [ OLDER (n keyword [, n keyword,... ] ) ]
  [ FUNCTION ( [ ALL ] [, ARCHIVE ] [, SEARCH ] [, RESTORE ]
     [, BROWSE ] [, DELETE ] [, SUBSET ] [, LOAD ]
     [, CONVERT ] [, IMPORT ] [, UPDATE ] ) ]
  [ PROCESS_MODE { ONLINE | BATCH } ]
  [ PROCESSED_BY userid ]
  [COMMIT_COUNT {n | 100}]
  [ TABLE [ cid. ] tblname ]
  [ COLUMN colname ]
  [ ARCHIVEDBY userid ]
  [ MISSINGFILE { NO | YES } ]
  [ EXPIRE { ( n keyword [, n keyword, ... ] ) |  explicitdate } ]
  [ ADNAME { group.user.name | NONE } ]
  [ FILE { NO | YES } ]
  [ UNEXP_RET_DATE { ERROR | WARNING } ]
```

All criteria keywords and operands for the DELETE statement are identical to those in the DIRECTORY statement. You can use these keywords and operands in a DIRECTORY statement to list and review candidates for deletion before submitting a job with the same keywords and operands in a DELETE statement.

Required Criteria Keywords

The keywords and values for operands are:

TYPE Type of objects to delete. TYPE must be included in the DELETE statement with one of the following operands.

ARC Delete Archive File entries that match other criteria.

ARCLOG
Delete Archive log entries that match other criteria.

NAME
Name identifying one or more Archive Files or Archive log entries to be deleted. This keyword is required.

filename
Name as an explicit value or a pattern, using DB2 LIKE syntax.

Note: Any EMC Centera file associated with the Archive File will also be deleted.

Optional Criteria Keywords

Use the following keywords as necessary to further refine the criteria for Archive File or log entries to be deleted.

For Archive File and Archive Log Entries

GROUP
Group designation for the Archive File referenced in Archive File or Archive log entries to delete.

group Group as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the Archive File referenced in the Archive File or Archive log entries to delete.

desc Description as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

FROMDATE
The oldest date in a range of Archive File creation dates. Archive File or Archive log entries for the selected Archive Files are deleted. You can use FROMDATE or OLDER, but not both.

yyyy-mm-dd
Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use FROMDATE alone to select all Archive Files created on or after that date.
- Use FROMDATE with TODATE to designate a range.

TODATE
The most recent date in a range of Archive File creation dates. Archive File or Archive log entries for the selected Archive Files are deleted. You can use TODATE or OLDER, but not both.

yyyy-mm-dd
Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use TODATE alone to select all Archive Files created on or before that date.
- Use TODATE with FROMDATE to designate a range.

OLDER

An interval used to retain Archive File or log entries, deleting all others; that is, only entries for files that are older than the specified interval are deleted. You can use OLDER, or FROMDATE and TODATE, but not both.

n The number of days, weeks, months, or years in the interval.

keyword

DAY(S)

WEEK(S)

MONTH(S)

YEAR(S)

You can use one or more *keywords* (DAY, WEEKS, etc.), each with an *n* operand, in any order.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

Examples:

- Specify the following on March 23, 1999, to delete all Archive Files created on or before March 23, 1998.

```
DELETE TYPE ARC NAME %
    OLDER (1 YEAR)
```

- Specify the following on June 30, 1999, to delete all Archive Files created on or before December 31, 1996.

```
DELETE TYPE ARC NAME %
    OLDER (2 YEARS, 6 MONTHS)
```

For Archive Log Entries Only

Use the following optional criteria keywords only when TYPE is ARCLOG. Using them when TYPE is ARC causes an error condition unless IGNORE_TYPE is specified as YES.

FUNCTION

Processes for which log entries are to be deleted.

ALL Delete all log entries.

ARCHIVE
Delete all Archive Process log entries.

SEARCH
Delete all Search Process log entries.

RESTORE
Delete all Restore Process log entries.

BROWSE
Delete all Browse Process log entries.

DELETE
Delete all Delete Process log entries.

SUBSET
Delete all Subset Process log entries.

LOAD
Delete all Load Process log entries.

CONVERT
Delete all Convert Process log entries.

IMPORT
Delete all Import Process log entries.

UPDATE
Delete all Update Process log entries.

PROCESS_MODE
The mode in which processes were executed. If the PROCESS_MODE keyword is not used, log entries are deleted without regard to mode.

BOTH Delete all log entries.

ONLINE
Delete log entries for online processes.

BATCH
Delete log entries for batch processes.

PROCESSED_BY
TSO ID of the user that initiated the processes for which log entries are deleted.

userid Specify any string that represents a User ID.

COMMIT_COUNT
Controls the frequency (number of archive files processed) for committing changes to the archive directory. The default is 100. Specifying a value less than the default may decrease the likelihood of locks being held on directory tables at the expense of processing performance.

Archive File Entries Only
Use the following optional criteria keywords only when TYPE is ARC. Using them when TYPE is ARCLOG causes an error condition unless IGNORE_TYPE is specified as YES.

TABLE
Name of a table referenced in Archive Files for which Archive File entries are deleted.

cid.tblname
The table name. Prefix with Creator ID, if needed to identify the table. Specify as an explicit value or a pattern, ending with %.

COLUMN
Name of a column referenced by objects in Archive Files for which Archive File entries are deleted.

colname
The column name. Specify as an explicit value or a pattern, using DB2 LIKE syntax.

ARCHIVEDBY
TSO ID of the user that created the Archive Files for the Archive File entries to be deleted.

userid Specify as an explicit value or a pattern, using DB2 LIKE syntax.

MISSINGFILE
File associations for deleted Archive File entries. Omit the MISSINGFILE keyword to delete Archive File entries without regard to the presence of a physical Archive File.

NO Delete Archive File entries without regard to the presence of an Archive File.

YES Delete only Archive File entries for which an Archive File is not present.

EXPIRE
Archive File expiration date for deleted Archive File entries.

Note: You may specify multiple *n keywords* within parentheses. If you specify one *n keyword* , you do not need parentheses.

n The number of days, weeks, months, or years to be added to the Archive File creation date. Any Archive File entries for Archive Files with an earlier expiration date are deleted.

keyword

DAY(S)

WEEK(S)

MONTH(S)

YEAR(S)

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

explicitdate

Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

ADNAME

Access Definition association for deleted Archive File entries. Omit the ADNAME keyword to delete Archive File entries without regard to the Access Definition.

group.user.name

Delete only Archive File entries for Archive Files created with the named Access Definition.

NONE

Delete only Archive File entries for Archive Files created with an unknown Access Definition. In other words, delete Archive File entries for Archive Files created with version 4.1 or earlier of Archive for DB2.

FILE Treatment of z/OS data sets for which Archive File entries are deleted.

NO Do not delete any z/OS data sets for which the Archive File entries are deleted (default).

YES Delete z/OS data sets for which the Archive File entries are deleted. If a z/OS file does not exist for an Archive entry, a warning message appears in the Summary Report.

UNEXP_RET_DATE

The action to be taken when the delete of an Archive File fails because the file retention date has not been reached. This keyword applies only when TYPE is ARC.

ERROR

Treat the retention date failure as an error condition (default value).

WARNING

Treat the retention date failure as a warning condition. The utility issues a warning message and continues processing with the next Archive File in the list.

Examples

The following are examples of DELETE statement usage:

1. To delete Archive File entries for all Archive Files older than one year, specify:

```
DELETE TYPE  ARC  NAME %  OLDER (1 YEAR)
```

2. To delete Archive File entries for all Archive Files with names prefixed by “FOPARC” that were created before July 1, 1998, specify:

```
DELETE TYPE  ARC  NAME  FOPARC.%
   TODATE 1998-07-01
```

3. To delete Archive Log entries for all processes initiated by user “FOPABC,” executed in batch mode before December 1, 2000, and involving files with names prefixed by FOPARC, specify:

```
DELETE  TYPE  ARCLOG NAME  FOPARC.%
   FUNCTION  ALL
   TODATE  2000-12-01
   PROCESSED_BY  FOPABC
   PROCESS_MODE  BATCH
```

ARCHIVE_IMPORT

At the time an Archive File is created, Archive places an entry in the Directory that enables Archive functions, such as Browse or Search, for the Archive File. However, if an Archive File is transferred, or moved to a different subsystem, an Archive entry is needed in the Directory in the new location or subsystem.

You can use an ARCHIVE_IMPORT statement to create an Archive entry subsequent to the creation of the Archive File. You must use a separate statement for each imported Archive File.

```
ARCHIVE_IMPORT
  NAME explicitfilename
  [ GROUP group ]
  [ DESC desc ]
  [ SECURITY { PUBLIC | PRIVATE | READONLY } ]
  [ SPARSE_INDEX ( [ cid. ] tblname, colname1 [, colname2, ... ] ) ]
  [ USE_EXISTING_INDEX dsname ]
  [ DENSE_INDEX ( [ cid. ] tblname, colname1 [, colname2, ... ] )
     { INDEX_FILE (  File Allocation Parameters ) |  INDEX_DSN  explicitfilename } ]
  [ REPLACE { NO | YES } ]
```

The keywords and operands in the ARCHIVE_IMPORT statement are:

Required Keyword

NAME
Name of the Archive File for which the Archive entry is to be created. This keyword and the *explicitfilename* operand must be included in the ARCHIVE_IMPORT statement and must precede any INDEX keywords.

explicitfilename
The fully qualified name of a cataloged Archive File data set.

Optional Parameter Keywords

GROUP
Group for the Archive File. If this keyword is omitted, the new Archive entry does not designate a Group for the associated Archive File.

group Value that is consistent with site requirements. If blank, the Archive File does not designate a Group.

DESC Description for the Archive File.

desc Description as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

SECURITY
Security status for the imported Archive File. This keyword is valid only if security is activated for the site. Otherwise, the keyword causes an error condition. Omit this keyword for site default.

PUBLIC
All can use and modify.

PRIVATE
Only the Archive Administrator or the owner can use and modify.

READONLY
All can use but only the Archive Administrator or the owner can modify.

SPARSE_INDEX
Include an index that consists of a range of values in the column. A sparse index occupies less space and can be searched more quickly than a dense index. You can have any number of sparse indexes for a table.

cid.tblname
The table name. Prefix with Creator ID, if needed to identify the table.

colname
The name of a column in the *cid.tblname* table to be indexed. The specified column must exist in the named table, or an error condition occurs.

USE_EXISTING_INDEX
Import the existing dense index along with the Archive File. (You may specify either the USE_EXISTING_INDEX or DENSE_INDEX keyword, but not both.)

dsname
Data set name of the dense index to be imported.

DENSE_INDEX
Index all values for the column. No more than 16 columns in a table may have dense indexes. Each value in a dense index can be no more than 254 bytes.

cid.tblname
The table name. Prefix with Creator ID, if needed to identify the table.

colname
The name of a column in the *cid.tblname* table to be indexed. The specified column must exist in the named table, or an error condition occurs.

INDEX_FILE
Designates the index file name and its allocation parameters. This keyword or INDEX_DSN is required when one or more DENSE_INDEX keywords is specified. The INDEX_DSN keyword may be used if the file already exists.

INDEX_DSN
Designates the index file name for dense indexes when one or more DENSE_INDEX keywords is specified.

explicitfilename
The fully qualified name of an existing VSAM data set. This VSAM file must be SMS-managed.

Optional Replace Keyword

REPLACE
Action taken if an Archive entry for the specified Archive File exists.

NO Skip this ARCHIVE_IMPORT statement.

YES Replace existing entry.

If you do not use the REPLACE keyword, the ARCHIVE_IMPORT statement is skipped when an Archive entry exists.

Example

To create an Archive entry for the Archive File PSTARC.CORP.OLDORDERS, with description and indexes for the ORDERS table columns CUST_ID and ORDER_ID, specify:

```
ARCHIVE_IMPORT NAME PSTARC.CORP.OLDORDERS
   DESC "1997 Orders"
   DENSE_INDEX (ORDERS, CUST_ID)
      INDEX_DSN PSTARC.CORP.INDEX
   SPARSE_INDEX (ORDERS, ORDER_ID)
```

ARCHIVE_UPDATE

Use an ARCHIVE_UPDATE statement to add indexes to an Archive entry or delete indexes from an Archive entry. You must use a separate ARCHIVE_UPDATE statement for each entry to be updated.

```
ARCHIVE_UPDATE
  NAME explicitfilename
  [ DROP_INDEX ( [ cid. ] tblname, colname1 [, colname2, ...] ) ]
  [ DROP_INDEX_TABLE ( [ cid. ] tblname1 [, [ cid. ] tblname2, ... ] ) ]
  [ DROP_INDEX_ALL { NO | YES } ]
  [ SPARSE_INDEX ( [ cid. ] tblname, colname1 [, colname2, ... ] ) ]
  [ DENSE_INDEX ( [ cid. ] tblname, colname1 [, colname2, ... ] )
     { INDEX_FILE (File Allocation Parameters) | INDEX_DSN  explicitfilename } ]
```

Keywords and operands in the ARCHIVE_UPDATE statement are

Required Keyword

NAME

Name of the Archive File for the Archive entry. This keyword and the *explicitfilename* operand must be included in the ARCHIVE_UPDATE statement and must precede other keywords in the statement.

explicitfilename

The fully qualified name of a cataloged Archive File data set.

Optional Functional Keywords

DROP_INDEX

Drop the index for one or more columns of a table in the named Archive File.

You can use multiple DROP_INDEX keywords for a table, but duplicate column names cause an error condition.

cid.tblname

The name of the table with the index to be dropped. An error condition occurs if Creator ID is not specified and there are multiple tables of the same name.

colname

The name of one or more columns in the table with an index to be dropped. If dropping indexes for multiple columns, provide the names in any order. Each specified column must exist in the named table and must have been indexed, or an error condition occurs.

DROP_INDEX_TABLE

Drop all indexes for the table. To replace dropped indexes, use the SPARSE_INDEX or DENSE_INDEX keyword after the DROP_INDEX_TABLE keyword.

cid.tblname

The name of the table with indexes to be dropped. An error condition occurs if Creator ID is not specified and there are multiple tables of the same name.

DROP_INDEX_ALL

Drop all indexes for all tables in the specified Archive File. To replace dropped indexes, use the SPARSE_INDEX or DENSE_INDEX keyword after the DROP_INDEX_ALL keyword.

NO Do not drop all indexes (default).

YES Drop all indexes.

SPARSE_INDEX

Include an index that consists of a range of values in the column. A sparse index occupies less space and can be searched more quickly than a dense index. You can have any number of sparse indexes for a table. However, duplicate column names cause an error condition.

cid.tblname

The table name. Prefix with Creator ID, if needed to identify the table.

colname

The name of a column in the *cid.tblname* table to be indexed. The specified column must exist in the named table, or an error condition occurs.

DENSE_INDEX

Index all values for the column. No more than 16 columns in a table may have a dense index. Each value in a dense index can be no more than 254 bytes. Duplicate column names cause an error condition.

cid.tblname

The table name. Prefix with Creator ID, if needed to identify the table.

colname

The name of a column in the *cid.tblname* table to be indexed. The specified column must exist in the named table, or an error condition occurs.

INDEX_FILE

Designates the index file name and its allocation parameters. This keyword or INDEX_DSN is required when one or more DENSE_INDEX keywords is specified and the Archive File does not have a dense index. The INDEX_DSN keyword may be used if the file already exists.

INDEX_DSN

Designates the index file name for dense indexes when one or more DENSE_INDEX keywords is specified.

explicitfilename

The fully qualified name of an existing VSAM data set. This VSAM file must be SMS⌂managed.

Example

Specify the following to add a dense index for the ORDERS table, CUST_ID column, in the Archive File PSTARC.CORP.OLDORDERS:

```
ARCHIVE_UPDATE  NAME PSTARC.CORP.OLDORDERS
   DENSE_INDEX  (ORDERS, CUST_ID)
      INDEX_DSN PSTARC.CORP.INDEX
```

OUTDD

An OUTDD statement indicates the DDNAME for the Directory listing data set. Each OUTDD statement applies until another OUTDD statement is encountered.

The default DDNAME for a Directory listing file, used if no OUTDD statement applies, is PSDFADIR.

`OUTDD` *ddname*

Example

The following is an example of an OUTDD statement to direct output to the data set corresponding to the DDNAME ADREPT:

```
OUTDD ADREPT
```

Utilities to Migrate Optim Objects

Optim provides Batch Utility control statements to migrate Optim objects stored in the Directory. You can export or import objects in batch or online.

Batch execution is documented in this section. (Online execution is documented in the *Common Elements Manual* under the heading Export and Import Optim Objects.) Batch processing allows you to migrate entire Directories, which can be useful when setting up a new subsystem. The online processes may be more useful when migrating small groups of objects.

Note: Batch export and import processing, formerly performed using JCLEXP and JCLIMP, is no longer supported. Use the Batch Utility control statements documented in this section to perform batch exporting and importing.

"EXPORT"
Select Optim objects in the Directory, by type and name of object, and store them in an external file. This file may then be used to import the objects to an Optim Directory in another DB2 subsystem.

"IMPORT" on page 48
Select Optim objects from a file and copy the objects to the Directory. This statement can also create Optim primary keys and relationships in the Directory from a file containing statements in DDL format (ALTER TABLE and CREATE TABLE).

"DEFCID" on page 50
Specifies the default Creator ID that may be used when processing DDL input for an IMPORT statement.

EXPORT

Use an EXPORT statement to select Optim objects in the Directory, by type and name of object, and copy the objects to a file. Any number of EXPORT statements may be used; however, after the execution of each EXPORT statement, all changes are committed.

For example, if the first EXPORT statement includes a delete, DB2 will commit the deletion after executing the EXPORT statement. This ensures that objects are deleted only after they have been exported. Objects are copied to the Export file in the order specified.

```
EXPORT
  TYPE { ALL | AC | AD | CD | CM | ED | LT | PK | PR | R | RD | RL | TM }
   NAME objectname
  CREATORID creatorid
  [ IGNORE_TYPE  { NO | YES } ]
  [ SUBDEF { NO | YES} ]
  [ DDNAME ddname ]
  [ DESC description ]
  [ TABLE [ cid. ] tblname ]
  [ COLUMN colname ]
  [ DELETE_OBJ { NO | SUBORD | YES  } ]
     [ SECURITY_VIOLATION { WARN | ERROR } ]
     [ SUBORD_NOTFND { WARN | ERROR } ]
```

To be exported, an object must satisfy criteria described by the selected keywords.

Required Criteria Keywords

TYPE Type of object to be exported. This keyword must be included in the EXPORT statement with one of the following:

ALL Export all objects, except Record Layouts, that match criteria.

AC Export all Archive Collections that match criteria.

AD Export all Access Definitions that match criteria.

CD Export all Compare Definitions that match criteria.

CM Export all Column Maps that match criteria.

ED Export all Environment Definitions that match criteria.

LT Export all Legacy Tables that match criteria.

PK Export all Primary Keys that match criteria.

PR Export all Column Map Procedures that match criteria.

R Export all Relationships that match criteria.

RD Export all Retrieval Definitions that match criteria.

RL Export all ***Data Ace*** Record Layouts that match name pattern.

Note: The resulting Legacy Tables are imported as TYPE LT.

TM Export all Table Maps that match criteria.

NAME

Name identifying the objects to be exported. This keyword and the *objectname* operand must be included in the EXPORT statement.

objectname

The name of the object to be exported as an explicit value or as a pattern using DB2 LIKE syntax.

- You can specify fewer than the number of levels allowed for the object type.
- A maximum of two levels (for example, PST%.%) may be specified if TYPE is ALL.

Record Layouts

CREATORID

Creator ID for Legacy Tables exported from ***Data Ace*** Record Layouts. This keyword and operand are required when TYPE is RL.

creatorid

The Creator ID for Legacy Tables. Resulting names for exported Legacy Tables are in the form *creatorid.name*, with *name* equal to the member name that matches the NAME keyword.

Optional Processing Keyword

IGNORE_TYPE

Indicate whether object types for which the specified criteria keywords do not apply should be exported. (Use IGNORE_TYPE only when TYPE is ALL.)

NO Export object types that match the specified criteria keywords in addition to all other object types.

YES Export only object types that match the specified criteria keywords (default).

If you do not use the IGNORE_TYPE keyword, the object is not exported.

Optional Criteria Keywords

Use the following keywords to further refine the criteria for objects to be exported.

SUBDEF

Export subordinate definitions. If no SUBDEF keyword is used, subordinate definitions are not exported.

NO Do not export subordinate definitions.

YES Export subordinate definitions.

DDNAME

Name of the DD statement for the output file. If no DDNAME keyword is provided, the default is PSDFEXPT.

ddname

DD Name as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the Optim objects to be exported.

desc Description as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double (“ ”) quotes.

TABLE

Name of a table referenced by objects to be exported.

cid.tblname

Table name as an explicit value or a pattern, using DB2 LIKE syntax. Prefix with Creator ID, if needed to identify the table.

COLUMN

Name of a column referenced by the object or in a table referenced by the object.

colname

Name of column.

DELETE_OBJ

After export, delete objects from Directory. If no DELETE_OBJ keyword is used, objects are retained.

NO Do not delete exported object.

SUBORD

Delete both exported object and subordinate objects.

YES Delete only the exported object.

SECURITY_VIOLATION

Error status if DELETE_OBJ fails. If no SECURITY_VIOLATION keyword is used, the error status is ERROR.

WARN

All objects that were successfully deleted are removed from the Directory.

ERROR

All objects that were successfully deleted are rolled back.

SUBORD_NOTFND

Error status if subordinate objects are not found.

WARN

All objects that were successfully deleted are removed from the Directory (default).

ERROR

All objects that were successfully deleted are rolled back.

Examples

The following are examples of EXPORT statement usage:

1. To export all primary keys for tables with the Creator ID "PSTDEMO," specify:

   ```
   EXPORT TYPE  PK  NAME  PSTDEMO.%
   ```

2. To export all relationships, specify:

   ```
   EXPORT TYPE  R  NAME  %.%.
   ```

3. To export all Access Definitions in the group "PSTDEMO" with the subordinate relationships and all Access Definitions for the user, Smith, in groups that begin with PST, without the subordinate relationships, specify:

   ```
   EXPORT  TYPE  AD  NAME  PSTDEMO.%.%
      SUBDEF YES
   EXPORT  TYPE  AD  NAME  PST%.SMITH.%
      SUBDEF NO
   ```

4. To export all Access Definitions, specify:

   ```
   EXPORT  TYPE  AD  NAME  %.%.%
   ```

5. To export all Record Layouts in the partitioned data set named in //PSDFDARL and prefix the names of resulting Legacy Tables with PSTDEMO, specify:

   ```
   EXPORT  TYPE  RL  NAME %
      CREATORID  PSTDEMO
   ```

IMPORT

Use an IMPORT statement to select Optim objects from a file created by an EXPORT statement and copy the objects to the Directory.

You can also use an IMPORT statement to create Optim primary keys and relationships from a file containing statements in DDL format (ALTER TABLE or CREATE TABLE). Any number of IMPORT statements may be used; objects are written to the Directory in the order specified.

Note: If the parameter ERROR CONT is used, it must precede all other parameters for the IMPORT process.

```
IMPORT
  TYPE { ALL | AC | AD | CD | CM | ED | LT | PK | PR | R | RD | TM }
  [ DDNAME ddname ]
  [ OVERWRITE { NO | YES } ]
  [ CREATE_GENERIC_REL { NO | YES } ]
  [ OWNERMODE { USER | SOURCE | EXPLICIT } ]
     [ OWNER ownerid ]
  [ CREATE_GENERIC_PK { NO | YES } ]
     [ PK_CREATORID { * | cid } ]
```

To be imported, an object must satisfy criteria described by keywords:

Required Criteria Keywords

TYPE Type of objects to be imported. This keyword must be included in the IMPORT statement with one of the following:

ALL Import all objects. If the definitions are in DDL format (ALTER TABLE or CREATE TABLE), TYPE ALL must be specified.

AC Import all Archive Collections that match criteria.

AD Import all Access Definitions that match criteria.

CD Import all Compare Definitions that match criteria.

CM Import all Column Maps that match criteria.

ED Import all Environment Definitions that match criteria.

LT Import all Legacy Tables, including exported ***Data Ace*** Record Layouts, that match criteria.

PK Import all Primary Keys that match criteria.

PR Import all Column Map Procedures that match criteria.

R Import all Relationships that match criteria.

RD Import all Retrieval Definitions that match criteria.

TM Import all Table Maps that match criteria.

DDNAME

Name of the DD statement defining the input file. If you do not provide a DDNAME keyword, the default is PSDFIMPT.

ddname

The DDNAME for the input file.

Optional Keywords

Use the following keywords to manage the Import Process.

All Objects

OVERWRITE

Action taken if an imported object has the same name as an object in the current Optim Directory. If you do not use the OVERWRITE keyword, an existing object is not replaced and an error is caused.

NO Ignore the conflicting imported object and proceed to the next.

YES Overwrite existing object in the Directory.

Relationships

CREATE_GENERIC_REL

Convert of relationships in the input file to generic relationships during the Import Process. If you do not use the CREATE_GENERIC_REL keyword, relationships are not converted.

NO Do not convert relationships to generic.

YES Convert all relationships to generic relationships.

Owner Processing

OWNERMODE

Determine the owner of all imported objects, as well as the date and time last modified. Specify:

USER The owner of all imported objects is the user performing the Import Process, and the date and time last modified is the current date and time (default).

SOURCE

The owner and the date and time last modified are set to the original values from the input file.

EXPLICIT

The owner of all imported objects is the value specified for the OWNER keyword. The date and time last modified is set to the original value from the input file.

Note: Only an administrator can specify SOURCE or EXPLICIT. Therefore, the PASSWORD statement must precede the IMPORT statement if OWNERMODE is SOURCE or EXPLICIT. For details, see PASSWORD.

OWNER
If OWNERMODE is EXPLICIT, specify the 1- to 8-character owner of all imported objects.

Primary Keys

CREATE_GENERIC_PK
Convert primary keys in the input file to generic Optim keys during the Import Process. If you do not use the CREATE_GENERIC keyword, primary keys are not converted.

NO Do not convert primary keys.

YES Convert primary keys to generic keys. You must also use the PK_CREATORID keyword.

PK_CREATORID
Convert primary keys if CREATE_GENERIC_PK is YES.

cid Creator ID of tables with primary keys to convert. Specify as an explicit value or a pattern, ending with %.

* Convert all primary keys.

DDL Input

If the input to the Import Process consists of DDL statements (CREATE TABLE or ALTER TABLE), you can optionally provide a DEFCID statement prior to the IMPORT statement. The DEFCID statement specifies a default Creator ID to be used when the DDL statement does not include a Creator ID with the table name. If the DDL statement does not include a Creator ID, and a DEFCID statement is not specified, an error condition occurs. (See “DEFCID” for more information.)

Examples

The following are examples of IMPORT statement usage:

1. To import all primary keys in PSDFIMPT and replace any primary keys with the same name, specify:
```
IMPORT  TYPE  PK
   OVERWRITE YES
```
2. To import all Access Definitions in PSDFIMPT, replace any Access Definitions with the same name, and indicate the owner is the original owner from the input file, specify:
```
PASSWORD  password
IMPORT  TYPE  AD
   OVERWRITE  YES
   OWNERMODE  SOURCE
```

Note: The PASSWORD statement must precede the IMPORT statement if OWNERMODE is SOURCE (or EXPLICIT). For details, see PASSWORD.

DEFCID

A DEFCID statement specifies the default Creator ID that may be used when processing DDL input for an IMPORT statement.

```
DEFCID creatorid
```

A DEFCID statement applies until another DEFCID statement is encountered.

Chapter 4. Processing Utilities

Processing utilities allow you to run Optim processes (such as Extract, Archive, Insert, Convert, Compare, Delete, Search, and Restore) in batch mode.

You can create batch jobs in the following two ways:

- Enter the job parameters directly in an editor, using the Batch Utility control statements.
- Use the Optim online panels to provide process request parameters and elect to run the request in batch. Optim executes the Batch Utility and uses the parameters you entered on the online panels to build the SYSIN data set, which consists of a series of control statements defining the job function to be performed.

In both cases, you can save the batch job and edit the control statements.

This section describes the following batch processing utilities:

"ARCHIVE" on page 52
Create an Archive File, using an Access Definition defined in the Optim Directory. You can use keywords to specify parameters similar to those for the online Archive Process and to override parameters in the Optim Directory Access Definition.

"COMPARE" on page 75
Compare two different sets of related data and create a Compare File to hold the results of the comparison. You can specify parameters similar to those for the online Compare Process.

"CONVERT" on page 83
Convert data from an Extract or Archive File to mask sensitive information or transform data to CSV format and create a converted file. You can specify parameters similar to those for the online Convert Process.

"DEFERRED_DELETE" on page 86
Delete archived data from the database, when deletion was deferred during the Archive Process. You can use keywords to provide parameters similar to those for the online Archive Delete Process.

"SEARCH" on page 91
Search for specific data within one or more Archive Files. The results are reported in a data set that can be reviewed online.

"RESTORE" on page 94
Select and restore archived data from Archive Files, using one or more Table Maps and optional criteria.

"SUBSET" on page 105
Create a Subset File that contains a referentially correct subset of rows from an existing Archive File.

"EXTRACT" on page 112
Creates an Extract File, using an Access Definition defined in the Optim Directory. You can use keywords to specify parameters similar to those for the online Extract Process and to override parameters in the Optim Directory Access Definition.

"INSERT" on page 125
Insert or update rows in DB2 tables using data from an Extract File. You can specify parameters similar to those for the online Insert Process.

“LOAD” on page 130
Transform the contents of an Extract File or an Archive File to Load Utility format. The output files created by this statement can then be used as input to an IBM load utility or another vendor's load utility to load the data into the database.

“CENTERA_OPTIONS” on page 135
Establish or override options for an Archive File stored on a Centera Server.

“TIVOLI _OPTIONS” on page 136
Establish or override options for an Archive File stored on an IBM Tivoli® Storage Manager Server.

“IGNORE_GENERIC_RELS” on page 137
Specifies whether generic relationships are ignored in process requests.

The syntax for each batch statement includes a number of keywords that allow flexibility and provide all of the options available online. For most processes, only a few keywords are needed: those to control processing and others to allocate an output file.

Keywords to control processing are shown in the syntax for each batch statement. For keywords to allocate an output file, see File Allocation Parameters.

Note: The batch processing utilities acknowledge all options that are specified on the online Site Options panel. However, User Options are not available to the utilities and are ignored. Control statements in which user options would be relevant include appropriate keywords to allow you to provide equivalent information.

ARCHIVE

Use an ARCHIVE statement to create an Archive File. An Archive File is a sequential file that contains the selected set of related rows from one or more tables and the object definitions for those tables. The Archive File is used as input to the Archive Restore and Browse Processes. Many users can access the Archive File repeatedly and simultaneously. The ARCHIVE statement requires an Access Definition defined in the Optim Directory. You can use parameters to override parameters in the Access Definition and to provide parameters similar to those for the online Archive Process.

The ARCHIVE statement provides several capabilities not available in the online Archive Process. You can generate Archive File and Index File names dynamically, based on the date and time. This is useful for automated scheduling of Archive Processes. Also, you can override the index parameters specified in the Access Definition.

Despite the large number of parameters allowed on the ARCHIVE statement, you can archive data using only two parameters, ACCESS_DEFINITION and ARCHIVE_FILE. For example, specify the following to create an Archive File named PSTUSER.ARCHIVE.ORDERS, using the Access Definition PSTUSER.AD.ORDERS:

```
ARCHIVE
   ACCESS_DEFINITION PSTUSER.AD.ORDERS
   ARCHIVE_FILE (DSNAME PSTUSER.ARCHIVE.ORDERS)
```

Use the ARCHIVE statement syntax as follows:

```
ARCHIVE
  { ACCESS_DEFINITION_DEFINE (parameters) ; | ACCESS_DEFINITION group.user.name
      [ DEFAULT_CID cid ]
      [ UNKNOWN { FAIL | ALLOW } ]
      [ SELECT ( [ cid. ] table1, coloperator, column1
         { [, critoperator, criteria ] | [ , DELETE ] }
         [, column2 { [, critoperator, criteria ] | [ , DELETE ] } ... ] ) ]
      [ SELECT ( [ cid. ] table2, coloperator, column1
         { [, critoperator, criteria ] | [ , DELETE ] }
```

```
       [, column2 { [, critoperator, criteria ] | [ , DELETE ] } ... ] ) ] ...
    [ SQL ( [ cid. ] table1 [, [/correlation/ ] whereclause1 ] ) ]
    [ SQL ( [ cid. ] table2 [, [/correlation/ ] whereclause2 ] ) ] ...
    [ VAR ( varname, value ) ]
    [ DATE ( [ cid. ] table1,
       column1, { [Yn] | [Mn] | [Wn] | [D n] | Xyyyy-mm-dd }
       [, column2, { [Yn] | [Mn] | [Wn] | [Dn] | Xyyyy-mm-dd } ... ] ) ]
    [ DATE ( [ cid. ] table2,
       column1, { [Yn] | [Mn] | [Wn] | [Dn] | Xyyyy-mm-dd }
       [, column2, { [Yn] | [Mn] | [Wn] | [D n] | Xyyyy-mm-dd } ... ] ) ] ...
    [ POINT_SHOOT_DSN explicitfilename ]
    [ POINT_SHOOT_ERROR { STOP | CONTINUE | SUBSET } ]
    [ OBJECT_DEFS { YES | NO } ]
    [ INCLUDE_OBJECT (obj1,obj2,...) ]
    [ EXCLUDE_OBJECT (obj1,obj2,...) ]
    [ ROW_SELECT_BY { ROWLIST | BOTH } ] }
ARCHIVE_FILE ( File Allocation Parameters )
[ DROP_INDEX_COLUMN ( [ cid. ] table, column1 [,  column2, ... ] ) ]
[ DROP_INDEX_TABLE ( [ cid. ] table1 [, [ cid. ] table2,... ] ) ]
[ DROP_INDEX_ALL { YES | NO } ]
[ SPARSE_INDEX ( [ cid. ] table, column1 [, column2, ...] ) ]
[ DENSE_INDEX ( [ cid. ] table, column1 [, column2, ... ] ) ]
    [ INDEX_FILE ( File Allocation Parameters )
    [ DUPE_INDEX_FILE (  File Allocation Parameters ) ] ]
[ UNLOAD_UTILITY { BMC | IBM | CDB | CDBO } ]
    [ IMAGE_COPY
       ( MODE { L | A | B | D }
       [ DATE yyyy-mm-dd ]
       [ TIME hh.mm.ss ]
       [ DSNAME dsname ] ) ]
    [ PARTITIONS ( part1 [, part2, ... ] )
       [ OTHER_PARTITIONS_SAME { YES | NO }] ]
[ DELETE_DEFER { YES | NO } ]
    [ CONTROL_FILE ( File Allocation Parameters ) ]
    [ DELETE_LOCK { YES | NO } ]
    [ DELETE_COMMIT_ROWS n ]
    [ DELETE_COMMIT_MINUTES n ]
    [ DELETE_DISCARD n ]
    [ ACTION_IF_NO_ROWS { WARNING | ERROR }
    [ RESTART { YES | NO } ]
    [ COMPARE_ROW { YES | NO } ]
    [ DEFAULT_KEY_LIMIT n ]
    [ ACCESS_METHOD ( cid.tablename, { K | S | E }, key limit ) ]
    [ DAA_TABLE ( tablename1, { YES | NO } ) ]
    [ DAA_TABLE ( tablename2, { YES | NO } ) ]
[ SECURITY_STATUS { PUBLIC | READONLY | PRIVATE } ]
[ GROUP group ]
[ DESC desc ]
[ ROW_LIMIT n ]
[ RETENTION_PERIOD { NOLIMIT | nD | nY | yyyy-mm-dd | yyyy.ddd | PERM | NEVER } ]
[ DUPE_ARCHIVE_FILE ( File Allocation Parameters ) ]
[ WITH_UR { YES | NO } ]
[ ACTION_COMMIT n ]
[ COLLECTION (cid.collectionname [, cid.collectionname2, ... ] ) ]
[ REPLACE_ARC_DIR { YES | NO } ]
[ SKIP_ARC_CATALOG { YES | NO | DUP } ]
[ REPORT_LEVEL { DETAIL | SUMMARY } ]
[ CENTERA_FILE { YES | NO }
    [ CENTERA_VALIDATE { START | DEFER } ] ]
[ TIVOLI_FILE { YES | NO }
    [ TIVOLI_VALIDATE { START | DEFER } ] ]
[ NOT_FOUND_RC4 { YES | NO } ]
[ EMPTY_TABLE_FOUND_RC4 { YES | NO } ]
[ DELETE_FAILED_ROWS_RC4 { YES | NO } ]
[ DELETE_0_ROWS_RC4 { YES | NO } ]
```

Access Definition parameters

Use the following parameters to identify the Access Definition and override or augment various parameters in the Access Definition.

ACCESS_DEFINITION

The name of the Access Definition. ACCESS_DEFINITION must be included in the ARCHIVE statement and must precede any parameters that override parameters in the Access Definition (e.g., SELECT, SQL, DEFAULT_CID).

group.user.name

The three-part Access Definition name.

ACCESS_DEFINITION_DEFINE

The Optim online process generates this parameter when it creates an Archive job for batch execution.

When you create an Archive job outside of the Optim online process, the best practice is to use the ACCESS_DEFINITION parameter to refer to a named Access Definition in the Optim Directory.

The ACCESS_DEFINITION_DEFINE parameters are within parentheses and a semicolon must follow the close parenthesis. (See ACCESS_DEFINITION_DEFINE Parameters for the allowable parameters.)

Note: The ADNAME parameter is generated when you specify a named Access Definition in the online process. This parameter is for documentation purposes only.

DEFAULT_CID

Override for the default Creator ID specified in the Access Definition. When used, this parameter must precede any parameters that require a default Creator ID (e.g., SELECT, DROP_INDEX_TABLE).

cid The default Creator ID.

UNKNOWN

The action to be taken if the Access Definition references unknown tables or relationships.

FAIL Terminate processing if any tables or relationships named in the Access Definition are unknown (default).

ALLOW

Bypass unknown tables and relationships and continue processing.

– Selection Criteria

SELECT

Selection criteria for rows in a table. Specify *critoperator* and *criteria* to augment Access Definition selection criteria (according to *coloperator*) or to override any existing Access Definition selection criteria for the specified column. Use DELETE, instead of *critoperator* and *criteria*, to bypass existing Access Definition selection criteria for the specified column.

cid.table

The table name. If you omit the Creator ID, the default Creator ID is used.

coloperator

The operator used to combine selection criteria for multiple columns in a table. You must specify one of the following:

AND Select data that matches criteria for all columns.

OR Select data that matches criteria for at least one column.

column The name of the column. You must specify at least one *column* operand.

critoperator
Operator for criteria. (See the following explanation for *criteria*.)

criteria Up to 250 characters of *criteria* per column. If the column has a character, graphic, or binary data type (e.g., CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE), the criteria must be delimited with single quotes. If the column has a numeric data type (e.g., INTEGER, SMALLINT, BIGINT, or DECIMAL), the criteria must not be in quotes.

Separate *critoperator* and *criteria* with a comma.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

- EQ, *criteria*
- NE, *criteria*
- GT, *criteria*
- LT, *criteria*
- GE, *criteria*
- LE, *criteria*
- IN, (*a,b,c,d...*)
- NOT IN, (*a,b,c,d...*)
- IS NULL
- IS NOT NULL
- LIKE, *pattern*
- NOT LIKE, *pattern*
- BETWEEN, *x* AND *y*
- NOT BETWEEN, *x* AND *y*

DELETE
Bypass existing selection criteria in the Access Definition for the column. If you use DELETE, you must omit *critoperator* and *criteria* for the column.

SQL An SQL WHERE clause for the specified table. Override any SQL WHERE clause specified in the Access Definition. (Omit *whereclause* to bypass an Access Definition SQL WHERE clause for the specified table.) You can specify multiple SQL statements, but duplicate table names cause an error.

cid.table
The table name. If you omit the Creator ID, the default Creator ID is used.

/correlation/
The optional correlation name must be enclosed in slashes and must immediately precede *whereclause*. Overrides any correlation name specified in the Access Definition.

whereclause
Lines of criteria as an SQL WHERE clause. Break each line at a blank, start anywhere on the next line, do not use quotes or parentheses, and end the clause with a close parenthesis.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

VAR Override for the default value of a substitution variable assigned in the Access Definition. If a default value was not defined in the Access Definition, you must use the VAR parameter to specify a value, or an error occurs.

varname
The name of the substitution variable assigned in the Access Definition. A colon (:) before *VarName* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

DATE Override for existing archive date criteria for the specified column in a table. (If the Access Definition does not include date criteria for the specified column, an error occurs.)

cid.table
The table name. If you omit the Creator ID, the default Creator ID is used.

column The name of the column. You must specify at least one *column* operand, with date criteria.

Y*n***, M***n***, W***n***, D***n***,** *or* **X***yyyy-mm-dd*
The cut-off date for archived data. Rows with an earlier date are selected for archiving. Specify an explicit date (in *yyyy-mm-dd* format), or the number of years, months, weeks, or days to subtract from the current date.

POINT_SHOOT_DSN
The name of a Point-and-Shoot file used to select data to be archived. Overrides any Point-and-Shoot file specified in the Access Definition

explicitfilename
The fully qualified name of the Point-and-Shoot File.

Note: You can also specify a Point-and-Shoot file by placing a PSDFPNS DD * data set in the job stream, containing key values to be processed. In this case, omit the POINT_SHOOT_DSN parameter.

POINT_SHOOT_ERROR
Indicate action to be taken when processing the Point-and-Shoot file results in errors.

STOP Stop the run if the Point-and-Shoot file contains invalid keys or is empty. This is the default.

CONTINUE
Continue the run if invalid keys are found, but do not use the Point-and-Shoot file.

SUBSET
If invalid keys are found, continue the run with the valid keys only. If errors other than invalid keys are found, stop the run.

– Object Definitions

OBJECT_DEFS
Indicate whether to include object definitions (e.g., primary keys, relationships, indexes, views, synonyms, aliases, procedures, triggers, and user-defined types and functions).

YES Include object definitions of all types (default).

NO Do not include object definitions.

INCLUDE_OBJECT
Include specific object types. The OBJECT_DEFS parameter must be NO. Specify any of the following values in any order within parentheses:

PKREL
Primary keys and relationships

INDEX
Indexes

VIEW Views

ALIAS
Aliases

SYN Synonyms

FPROC
Column Field Procedure Names

TRIG Triggers

SPROC
Stored Procedures

UDEF User Defined Types and Functions

EXCLUDE_OBJECT
Exclude specific object types. The OBJECT_DEFS parameter must be YES. Specify any of the following values in any order within parentheses:

PKREL
Primary keys and relationships

INDEX
Indexes

VIEW Views

ALIAS
Aliases

SYN Synonyms

FPROC
Column Field Procedure Names

TRIG Triggers

SPROC
Stored Procedures

UDEF User Defined Types and Functions

ROW_SELECT_BY
Indicate whether selection criteria or Point-and-Shoot list should be used when Access Definition specifies both for the Start Table.

ROWLIST
Apply Point-and-Shoot list only (default).

BOTH Apply Point-and-Shoot list and selection criteria.

Archive File parameter

ARCHIVE_FILE
Use the parameters in File Allocation Parameters to name and allocate the Archive File.

Index parameters

Use the following parameters to override index specifications in the Access Definition.

DROP_INDEX_COLUMN

Drop the index for one or more columns in a table in the Access Definition. (If the Access Definition does not have an index for the specified column, an error occurs.) You can specify multiple DROP_INDEX_COLUMN parameters, but duplicate column names cause an error.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns with an index to be dropped.

DROP_INDEX_TABLE

Drop all indexes for the specified tables in the Access Definition. You can specify multiple DROP_INDEX_TABLE parameters, but duplicate table names cause an error.

cid.table
The name of one or more tables with indexes to be dropped. If you omit the Creator ID, the default Creator ID is used.

DROP_INDEX_ALL

Drop all indexes for all tables in the Access Definition.

YES Drop all indexes.

NO Do not drop all indexes (default).

SPARSE_INDEX

Create a sparse index for one or more columns in the specified table. If the Access Definition includes a sparse index for the specified column, an error occurs unless the index has been dropped using a prior DROP_INDEX parameter.

You can use multiple SPARSE_INDEX parameters, but duplicate column names cause an error. You can specify a sparse index for a column that has a dense index.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns to be indexed.

DENSE_INDEX

Create a dense index for one or more columns in the specified table. If the Access Definition defines a dense index for the specified column, an error occurs unless the index has been dropped using a prior DROP_INDEX parameter.

You can use multiple DENSE_INDEX parameters, but duplicate column names cause an error. You can create a dense index for a column that also has a sparse index.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns to be indexed.

INDEX_FILE

Create an Index File for dense indexes. If omitted, and a dense index is to be created, an error occurs. Use the parameters shown in File Allocation Parameters.

DUPE_INDEX_FILE

Name and allocate a duplicate Index File. This parameter is allowed only if the INDEX_FILE parameter has been specified. Use the parameters shown in File Allocation Parameters.

Unload Utility parameters

Use the following parameters to unload data:

UNLOAD_UTILITY
The name of the unload utility used to retrieve the data. If omitted, DB2 is used to retrieve the data. This parameter is required to extract data from an image copy.

BMC Use the BMC UNLOAD PLUS utility.

IBM Use the High Performance Unload utility.

CDB Use the CDB Auto-Unload utility.

CDBO
Use the CDB Auto-Online Unload utility.

Note: You must add the appropriate utility DD statements. Refer to the utility documentation for the required DD statements.

IMAGE_COPY
The image copy file to use as input. If omitted, DB2 files are used as input. This parameter is allowed only if the UNLOAD_UTILITY parameter is specified.

IMAGE_COPY parameters must be enclosed in parentheses. MODE is required; other parameters are optional, depending on the value of MODE.

Note: To archive data from DB2 image copy data sets in multiple partitions of the same tablespace stored on the same tape volume, you must manually edit the JCL to allocate the data sets. Multiple image copy data sets cataloged on the same tape volume can not be allocated using dynamic allocation. This is a z/OS limitation. If you attempt to use dynamic allocation, the archive process fails with a dynamic allocation error. Refer to Chapter 5, “File Allocation Parameters,” on page 139 for details.

MODE
The image copy file to use:

L Use the latest file.

A Use the first file created on or after the specified DATE and TIME.

B Use the first file created on or before the specified DATE and TIME.

D Use the file named in DSNAME.

DATE The date criteria for the image copy file in this format: *yyyy-mm-dd*.

TIME The time criteria for the image copy file in this format: *hh.mm.ss.*If omitted, a value of 00.00.01 is assumed

DSNAME
The fully qualified name of the image copy file.

PARTITIONS
The numbers of the partitions to be processed. If omitted, all partitions are processed. This parameter applies only if the UNLOAD_UTILITY parameter is specified and the Start Table is partitioned.

part The partition numbers, enclosed in parentheses and separated by commas.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

You may specify the partitions in any order. (Partitions that are not specified are not archived.)

OTHER_PARTITIONS_SAME
Use this operand to indicate whether to process tables in all partitions or only the tables in the partitions specified in the PARTITIONS operand for the Start Table.

YES Limit processing to the tables in the partitions specified in the PARTITIONS operand for the Start Table. This assumes that the rows selected from related tables are physically located in the same partitions as the rows selected from the Start Table.

NO Do not limit processing to the tables in partitions specified in the PARTITIONS operand for the Start Table. If needed, process tables in additional partitions to select rows from related tables. This is the default.

Delete After Archive parameters

Use the following parameters to provide processing options for the delete after archive operation. Note that these parameters apply only if the Access Definition contains delete after archive specifications.

DELETE_DEFER
Indicate whether the Delete Process should be deferred.

YES Defer the Delete Process. All other delete after archive parameters are ignored and you must use the DEFERRED_DELETE statement to perform the Delete Process. See "DEFERRED_DELETE" on page 86 for more information.

NO Do not defer the Delete Process (default).

CONTROL_FILE
Name of the Control File and its allocation parameters. This parameter is required to execute the Delete Process. Use the parameters shown in File Allocation Parameters.

DELETE_LOCK
Indicate whether to lock tables during the Delete Process. Locking tables ensures other database activity does not interfere with the Delete Process. However, it will prevent other users from accessing the table.

YES Lock tables. (This setting causes an error if site options prevent a user from locking tables.)

NO Do not lock tables (default).

DELETE_COMMIT_ROWS
The frequency of commits in rows for the Delete Process. The commit points affect the starting point in case of a Restart. Frequent commits keep page locks to a minimum. If you omit both DELETE_COMMIT_ROWS and DELETE_COMMIT_MINUTES, the site limit is used.

n Commit each *n* number of rows where *n* is a value from 1 through the Site Options Commit Frequency Rate, inclusive.

Note: If DELETE_LOCK is set to YES, DELETE_COMMIT_ROWS and DELETE_COMMIT_MINUTES are ignored. A commit is performed as processing for each table is completed.

DELETE_COMMIT_MINUTES
The frequency of commits in minutes for the Delete Process. If you omit DELETE_COMMIT_MINUTES, DELETE_COMMIT_ROWS determines the commit frequency.

n Commit every *n* minutes, where *n* is 1 through 1440.

DELETE_DISCARD

The maximum number of rows that can be discarded when deleting rows, before terminating the process. If the limit is met, the process is terminated. You can use Restart to begin the Delete Process again at the termination point

n Specify a value in the range 1 to 4,294,967,295.

blank No limit on discarded rows.

ACTION_IF_NO_ROWS

The action to be taken if no rows are deleted from the database. That is, should the utility generate a warning or an error condition?

WARNING

Treat no rows being deleted as a warning condition. The utility issues a warning message and continues processing (default).

ERROR

Treat no rows being deleted as an error condition.

RESTART

Indicate whether to restart or retry the delete portion of an Archive Process if a processing failure occurs. Optim automatically determines whether to perform a restart or a retry.

YES Restart or retry the delete, and, optionally, respecify any DELETE parameters.

NO Do not attempt to restart or retry the delete (default).

Note: You can also restart or retry the process by adding the RESTART operand to the PARM field on the batch EXEC statement. Use a blank to separate the RESTART operand from the previous operand in the PARM field.

COMPARE_ROW

Indicate whether the Delete Process compares rows of data in the Archive File with rows in the database prior to deletion.

YES Rows are deleted from the database only if they exactly match rows in the Archive File. Rows that do not exactly match are discarded and noted in the Control File. (This setting causes an error if the **Compare Row Contents** site option is not set to USER.)

NO Row comparison is not performed (default). This may improve performance significantly; however, you risk losing any updates to the data in the database since the Archive Process was performed.

DEFAULT_KEY_LIMIT

Specify the default maximum number of keys to be processed at one time when deleting rows for any table in the Archive File. Applies when processing a table for which a key limit is not explicitly defined in the Access Definition or using the ACCESS_METHOD parameter.

n 1 - 100 (default is 1)

The following conditions must be true to process multiple keys at one time when deleting rows:

- An index on the primary key is defined for the table.
- COMPARE_ROW parameter is NO.
- Row-level Archive Actions are not defined for the Delete Process (e.g., Before Delete of Row).
- The table is not a parent in a DBMS relationship.

ACCESS_METHOD

Indicate how to access rows to be deleted from tables in the Archive File. You may specify this parameter once for each table to be processed. If you omit it for a table, Optim determines how to access the rows in that table. You must specify the parameters within parentheses, separated by commas, and in the following order.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

cid.tablename

The table name

K Use key lookup

S Use table scan

E Optim determines the access method.

key limit

Maximum number of keys to be specified in an SQL statement when the access method is K. Specify a number between 1 and 100. You may omit this parameter if the access method is S.

Note: If you used the online process to generate the Archive batch job, you must use the ACCESS_METHOD parameter to override the default access method (that is, the Optim-determined access method).

DAA_TABLE

Indicates whether data from a table in the Access Definition is to be deleted. DAA_TABLE overrides any value from the Access Definition. If a DAA_TABLE value is not specified, the DAA value for each table is taken from the Access Definition. Multiple parameters are permitted as long as the table name is not repeated.

If you specify DAA_TABLE with the DELETE_DEFER parameter, DAA_TABLE will apply to the subsequent DEFERRED_DELETE statement unless you override it with another DAA_TABLE parameter in the DEFERRED_DELETE statement.

tablename

Name of the table.

YES Delete the table data.

NO Do not delete the table data.

Processing parameters

Use the following parameters to specify processing options for the Archive File.

SECURITY_STATUS

Security status to be assigned to the Archive File. If archive security is not currently activated, the assigned status becomes effective when security is activated for the site. Omit this parameter to use the site default for archive security (or PUBLIC, if the site default is not specified).

PUBLIC

All can use and modify.

READONLY

All can use, but only the Archive Administrator or the owner can modify.

PRIVATE

Only the Archive Administrator or the owner can use and modify.

GROUP

Group designation for the Archive File.

group Up to 8-character group designation, enclosed in single or double quotes if using blanks or special characters.

DESC Description of the Archive File.

desc Up to 40-character description, enclosed in single or double quotes.

ROW_LIMIT

The maximum number of rows that can be archived. Specify a value in the range 1 to 4,294,967,295. If omitted, the site limit is used.

n The maximum number of archived rows.

RETENTION_PERIOD

The retention period for the Archive File. You cannot delete or overwrite the Archive File, its associated Archive Directory entry, or any associated Archive Index File until after the specified retention period.

This parameter is valid only when the Site Option, **Specify Arc File Retention**, is set to User. If omitted, when the Site Option allows user specification, no retention period is assigned.

NOLIMIT

No retention period is assigned.

*n***D** The number of days to retain the Archive File.

*n***Y** The number of years to retain the Archive File.

*yyyy***-***mm***-***dd*, *yyyy/mm/dd*, **or** *yyyy.ddd*

An explicit date, after which you can delete or overwrite the Archive File. Enter a year from 1900-2155. For *ddd* values, enter a day from 000-366.

PERM, NEVER, 1999.365, 1999.366,
1999/12/31, or 99/12/31

The retention period does not expire.

Note: Any Archive File assigned a retention period over 9999 days or 27 years is considered permanent, and can only be deleted or overwritten if you reduce the retention period using the ALTER statement of the IBM Utility program, IDCAMS.

Duplicate Archive File parameter

DUPE_ARCHIVE_FILE

Name and allocate a duplicate Archive File. Use the parameters shown in File Allocation Parameters. DUPE_ARCHIVE_FILE has the same parameters and operands as ARCHIVE_FILE. However, you can omit the TAPE parameter, since the duplicate Archive File TAPE value is automatically set to the value for the Archive File.

– Miscellaneous Options

WITH_UR

Indicate whether to archive uncommitted data from the database. This parameter is valid only when the Site Option, **Use Uncommitted Reads**, is set to User. Specify:

YES Archive uncommitted data from the database.

Note: If you choose to archive uncommitted data, the relational integrity of the data in the Archive File may be compromised. Use caution if restoring data in any Archive File with uncommitted data.

NO Do not archive uncommitted data from the database. This is the default.

ACTION_COMMIT
The commit interval for a table for which archive actions are specified. If omitted, commits are not performed for archive actions.

n The number of archive actions executed between commit operations. Specify a value in the range 1 - 4,294,967,295.

COLLECTION
If appropriate, specify the name of the Archive Collection(s) to which the Archive File should be added. Each collection is identified by a Collection ID of up to eight characters, and a Collection Name of up to 12 characters, such as CID1.COLLECTION. You can specify multiple collections, but duplicate names cause an error.

cid The Collection ID.

collectionname
The Collection Name.

REPLACE_ARC_DIR
The action taken if an Archive Directory entry already exists for the specified Archive File and the duplicate Archive File, if one is requested.

YES Replace the existing entry.

NO Stop the Archive process (default).

SKIP_ARC_CATALOG
Indicate whether to create an Archive Directory entry for the Archive File and the duplicate Archive File, if one is requested.

YES Skip creating an entry for the Archive File or the duplicate Archive File

NO Create an entry for the Archive File and the duplicate Archive File, if one is specified (default).

DUP Skip creating an entry for the duplicate Archive File.

REPORT_LEVEL
The level of detail provided in the Process Report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

– Centera Options

CENTERA_FILE
Indicate whether to create a Centera file.

YES Create a Centera file.

Note: If you enter YES and a CENTERA_OPTIONS statement has been specified, those options will override Centera Site Options specifications.

NO Do not create a Centera file (default).

CENTERA_VALIDATE
Indicate when the Centera Pool is validated.

START
The Centera Pool is validated before data is archived (default). If the pool is invalid, the Archive Process is terminated.

DEFER
The Centera Pool is validated after the data is archived.

If the pool is invalid, the Archive File can be copied to Centera using the Archive Files panel. For more information about copying files to Centera using the Archive Files panel, see the *Archive User Manual*.

– Tivoli Options

TIVOLI_FILE
Indicate whether to create a Tivoli file.

YES Create a Tivoli file.

Note: If you enter YES and a TIVOLI_OPTIONS statement has been specified, those options will override Tivoli Site Options specifications.

NO Do not create a Tivoli file (default).

TIVOLI_VALIDATE
Indicate when the Tivoli Alias is validated.

START
The Tivoli Alias is validated before data is archived (default). If the alias is invalid, the Archive Process is terminated.

DEFER
The Tivoli Alias is validated after the data is archived.

If the alias is invalid, the Archive File can be copied to Tivoli using the Archive Files panel. For more information about copying files to Tivoli using the Archive Files panel, see the *Archive User Manual*.

NOT_FOUND_RC4
Indicate the action taken and return code (RC) setting when rows have not been found during Delete batch processing.

Note: If the source file has no rows, Optim ignores this parameter and issues a warning and RC=8 or RC=4, depending on the specification for ACTION_IF_NO_ROWS.

YES A warning message will be issued and the program will end with RC=4 (default).

NO The program will end with RC=0.

EMPTY_TABLE_FOUND_RC4
Indicate the action taken and return code (RC) setting when empty tables have been detected during Delete batch processing.

Note: If the source file has no rows, Optim ignores this parameter and issues a warning and RC=8 or RC=4, depending on the specification for ACTION_IF_NO_ROWS.

YES Empty tables will be processed, a warning message will be issued, and processing will end with RC=4.

NO Empty tables will be ignored and processing will end with RC=0 (default).

DELETE_FAILED_ROWS_RC4
Indicate the action taken and return code (RC) setting when some rows of a table have failed to be deleted because a condition existed that prevented it (for example, a violation of Referential Integrity would occur).

YES A warning message will be issued and processing will end with RC=4.

NO Processing will end with RC=0 (default).

DELETE_0_ROWS_RC4

Indicate the action taken and return code (RC) setting during Delete batch processing when a table has rows in an Archive File but no rows have been deleted. Possible reasons include

- The rows cannot be deleted because there are no matching rows in DB2 tables
- A condition existed that prevented it (for example, a violation of Referential Integrity would occur).

YES A warning message will be issued and processing will end with RC=4.

NO Processing will end with RC=0 (default).

Example

Use the following statement to create an Archive File named OPTUSER.ARCHIVE.CUST, using the OPTUSER.AD.CUSTOMERS Access Definition. This example also uses selection criteria, executes the UNLOAD PLUS utility, and provides a description for the Archive File.

```
ARCHIVE
   ACCESS_DEFINITION  OPTUSER.AD.CUSTOMERS
   ARCHIVE_FILE (DSNAME OPTUSER.ARCHIVE.CUST,
      MODE  REP, TAPE YES)
   SELECT (OPTUSER.CUSTOMERS, AND, AGE, GR, 21, AREA,
      EQ, 'WEST')
   UNLOAD_UTILITY BMC
   DESC "WESTERN CUSTOMERS OVER 21 YEARS"
```

ARCHIVE Override Keywords

You can use override keywords to defer the execution of delete after archive specifications or replace the default Creator ID, selection criteria, and SQL WHERE clause in the Access Definition.

Use the PSDFOVRD DD statement to provide the desired overrides. These overrides are intended to be used only in a batch Archive Process. Do not use them in any other process, including a Deferred Delete Process. (Examples follow the discussion of the overrides.)

Archive Date Criteria

To replace archive date criteria in the Access Definition, specify:

```
ARCHIVE_DATE_YEAR [cid.]table column value
ARCHIVE_DATE_MONTH [cid.]table column value
ARCHIVE_DATE_WEEK [cid.]table column value
ARCHIVE_DATE_DAY [cid.]table column value
ARCHIVE_DATE_EXPLICIT [cid.]table column yyyy-mm-dd
```

cid.table
Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID in the Access Definition is used.

column Column name must be specified. If the column does not have date criteria specified, an error occurs.

value Value is required. Value must be a positive number greater than zero.

ARCHIVE_DEFER_DELETE

To replace the **Defer Delete After Archive** specification, specify:

```
ARCHIVE_DEFER_DELETE     { Y | N }
```

Use the Y operand to defer any delete after archive specifications in the Access Definition. Use N to delete, according to the Access Definition specifications, when the Archive File is created.

COMPARE_ROW

To set the indicator for comparing rows before deleting them, specify:

```
COMPARE_ROW     { Y | N }
```

This override allows you to determine whether rows are compared before deleting. (Use this keyword only if the **Compare Row Contents** Site option is set to USER.)

ARCHIVE_DSN

To specify a new data set name for the Archive File, specify:

```
ARCHIVE_DSN     data.set.name
```

This override allows you to use one set of saved JCL to create Archive Files with different names.

ARCHIVE_DSN_DUP

To specify a new data set name for the duplicate Archive File, specify:

```
ARCHIVE_DSN_DUP     data.set.name
```

This override is valid if duplicate files are active according to a Site Option.

ARCHIVE_INDEX_DSN

To specify a new data set name for the Archive Index File, specify:

```
ARCHIVE_INDEX_DSN     data.set.name
```

This override allows you to use one set of saved JCL to create Archive Files and Index Files with different names.

ARCHIVE_INDEX_DSN_DUP

To specify a new data set name for the duplicate Archive Index File, specify:

```
ARCHIVE_INDEX_DSN_DUP     data.set.name
```

This override is valid if duplicate files are active according to a Site Option.

UNL_IMAGECOPY_DSN

To override the Image Copy DSN parameter for an unload program, specify:

```
UNL_IMAGECOPY_DSN     image.file.dsn
```

UNL_IMAGECOPY_DATE

To override the Image Copy Date parameter for an unload program, specify:

```
UNL_IMAGECOPY_DATE     yyyy-mm-dd
```

UNL_IMAGECOPY_TIME

To override the Image Copy Time parameter for an unload program, specify:

```
UNL_IMAGECOPY_TIME     hh.mm.ss
```

UNL_IMAGECOPY_SELECT

To override the Image Copy Criteria parameter for an unload program, specify:

```
UNL_IMAGECOPY_SELECT     { A | B | L | S }
```

A First Image Copy file created on or after the specified Date and Time.

B First Image Copy File created on or before the specified Date and Time.

L Latest Image Copy file. Any Date and Time values are ignored.

S Image Copy file. The name is provided as the Image Copy DSN parameter.

UNL_OBID

To process an Image Copy data set created on a different subsystem, specify:

```
UNL_OBID    [cid.]tblname  obid
```

cid.tblname
The table name must be specified. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

obid The DB2 Object Identifier must be specified and is used to generate the OBID parameter or the ORIGINOBID parameter on the UNLOAD statement. See the appropriate BMC or IBM reference manual for more information.

COMMIT_COUNT

To override the commit count for deletes that was specified when the job was created, specify:

```
COMMIT_COUNT      value
```

The value can range from zero to 4,294,967,295.

COMMIT_MINUTES

To change commit processing from number of deletes to elapsed time, specify:

```
COMMIT_MINUTES      value
```

The value is specified in minutes and will override the commit count. The value can range from 1 to 1440. The process report will reflect the change from the number of updates to elapsed time.

ARCHIVE_ACTION_COMMIT

To override the value for the commit interval for archive actions that was specified when the job was created, specify:

```
ARCHIVE_ACTION_COMMIT      value
```

The value can range from 0 to 4,294,967,295. If no commit interval was specified previously, an override value greater than zero will activate the commit interval. If you specify an override value of zero, commits will not be performed for archive actions.

DEFCID

To override the default Creator ID in the Access Definition, specify:

```
DEFCID     cid
```

cid The default Creator ID. This Creator ID applies only to tables that are not explicitly qualified in the Access Definition.

This override also affects the names of the tables in the relationships on the Relationship Usage list. If a relationship is not found for the updated table name, an error occurs when the archive is performed. Only one DEFCID parameter may be specified for an Archive Process.

Selection Criteria

To replace or bypass selection criteria in the Access Definition, specify:

```
SEL    [cid.]table   colname   [ selcriteria ]
```

cid.table
Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

colname
Column name is required.

selcriteria
Selection criteria. Specify replacement criteria or leave blank to ignore Access Definition criteria for the current process. Selection criteria must conform to the format required on the Specify Selection Criteria for AD panel (see the *Common Elements Manual* for further information). A replacement specification cannot exceed 53 characters.

Selection criteria can be specified for one or more columns in the table. However, each must apply to a different column. You can specify only one SEL parameter for each column in a table.

SQL

To override the SQL WHERE clause in the Access Definition for a table or to specify an SQL WHERE clause for a table that does not have one in the Access Definition, specify:

```
SQL   [cid.]tblname  [ /correlation/ ]  [ where ]
```

cid.tblname
The table name must be provided. If you do not provide the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

/correlation/
Add or change a correlation name. If you specify a correlation name, it must immediately follow *tblname* and be enclosed in slashes.

where The SQL WHERE clause. The clause must conform to the requirements specified in the *Common Elements Manual* in the topic **SQL WHERE Clause Specifications**. However, the keyword WHERE is not required.

If you do not specify an SQL WHERE clause, any SQL WHERE clause specified in the Access Definition is ignored for the current Archive Process.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

You can specify the WHERE Clause override for more than one table as long as a separate SQL parameter is provided for each. You can specify only one SQL parameter for a table.

EVERY_NTH_ROW

To override the numeric value in the Access Definition used as a factor for selecting rows from a table, specify:

```
EVERY_NTH_ROW    [cid.]tblname   value
```

cid.tblname

Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

value A numeric value to specify a sampling factor for a table (*tblname*). Valid values are 1 through 65,535.

ROW_LIMIT

To override the numeric value in the Access Definition used to limit the number of rows selected from a table, specify:

```
ROW_LIMIT    [cid.]tblname   value
```

cid.tblname

Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

value A numeric value to limit the number of rows selected from a table (*tblname*). Valid values are 1 through 4,294,967,295.

VAR

To override the default value of a substitution variable assigned in the Access Definition, specify:

```
VAR    varname   value
```

varname

The name of the substitution variable assigned in the Access Definition. A colon (:) in front of *VarName* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP or TIMESTAMP WITH TIME ZONE column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

UNKNOWN

To override the default treatment for objects referred to in the Access Definition that no longer exist, specify:

```
UNKNOWN    { FAIL | ALLOW }
```

FAIL Terminate the Archive Process if any tables or relationships named in the Access Definition have become unknown since the batch process was initiated. The report contains a message noting the first unknown object. FAIL is the default.

ALLOW

Bypass the unknown tables and relationships and continue with the Archive Process. The report contains a message listing the unknown objects.

Tables and the relationships between the tables used in the Archive Process to traverse the database must exist. However, be aware that bypassing an unknown table or relationship may also exclude “related” tables listed in the Access Definition even when these tables exist.

UNKNOWN ALLOW is useful when you use DEFCID to override the default Creator ID. The change in the default Creator ID may result in names of tables that do not exist. Use the UNKNOWN ALLOW override to direct the Archive Process to skip these “unknown” tables.

REPLACE_ARCHIVE_DIR

To override a User option governing the replacement of Archive Directory entries in a batch Archive Process, specify:

```
REPLACE_ARCHIVE_DIR    { Y | N }
```

Use the Y operand to replace any existing Directory entry. Use N to cause an error when a Directory entry for the Archive File exists.

CENTERA_VALIDATE

To override validation of the Centera Pool at the beginning of the Archive Process, specify:

```
CENTERA_VALIDATE  DEFER
```

If validation is done at the start of the Archive Process and the pool is invalid, the process is terminated. If validation is deferred and the pool is invalid, the Archive File can be copied to Centera using the Archive Files panel. See the *Archive User Manual* for further information.

TIVOLI_VALIDATE

To override validation of the Tivoli Alias at the beginning of the Archive Process, specify:

```
TIVOLI_VALIDATE  DEFER
```

If validation is done at the start of the Archive Process and the alias is invalid, the process is terminated. If validation is deferred and the alias is invalid, the Archive File can be copied to Tivoli using the Archive Files panel. See the *Archive User Manual* for further information.

WITH_UR

To override the archiving of uncommitted data from the database during the Archive Process, specify:

```
WITH_UR    { Y | N }
```

Use the Y operand to archive uncommitted data from the database. Use N to only archive committed data.

Note: This override is only available if the **Use Uncommitted Reads** Site option is set to U.

Rules for Parameters

The following rules apply when specifying these overrides:

- One or more overrides may be included in the JCL.
- Each override keyword must begin in the first space of the line.
- The qualifiers for the overrides must be separated by a space.
- If an override spans multiple lines, continue on the next line.
- You can comment the list by specifying an asterisk, *, in the first position of each comment line.
- If multiple selection criteria are specified, they are ANDed or ORed depending on the type of Archive processing:
  - For batch execution of an Archive process generated online, criteria are combined according to your response on the Specify Selection Criteria panel.
  - For the Batch Utility Archive statement, criteria are combined according to the coloperator value of the SELECT keyword.
- If both an SQL WHERE clause and selection criteria are specified for an individual table, the clauses are ANDed.

- You can store the overrides in a sequential file or a partitioned data set. If record length exceeds 80, only the first 80 characters are processed. Sequence numbers are not allowed.
- The overrides must conform to the syntax required when specified directly in the jobstream.

Override Examples

1. To override the default Creator ID and specify selection criteria for two of the tables in the archive, insert the following in the JCL:

   ```
   //PSDFOVRD  DD  *
   DEFCID   PSTDEMO2
   * LIMIT SELECTION TO CUSTOMERS IN NEW JERSEY
   * WHO HAVE ORDERS FOR WHICH THE
   * FREIGHT CHARGES EXCEEDED $50.00
   SEL   CUSTOMERS  STATE  ='NJ'
   SEL   ORDERS  FREIGHT_CHARGES  >50.00
   ```

   You must include one or more spaces between the column name and the selection criteria.
2. To specify a sequential file named PST.SAMPLE.PARMS as the source of the parameter list, insert in the JCL:

   ```
   //PSDFOVRD  DD  DSN=PST.SAMPLE.PARMS,DISP=SHR
   ```
3. To use an Image Copy file named PST.IMAGE.COPY in an Archive Process utilizing an unload program, insert in the JCL:

   ```
   //PSDFOVRD  DD  *
   * CHANGE IMAGE COPY SPECIFICATION TO
   * A PARTICULAR DATASET
   UNL_IMAGECOPY_SELECT    S
   UNL_IMAGECOPY_DSN   PST.IMAGE.COPY
   ```

Offline Delete Archive Process

A batch Delete Archive Process allows you to avoid the overhead of DB2 logging, an important consideration when deleting large quantities of data from your database.

This offline Delete Archive Process has the same effect as deleting “delete after archive” data directly from the database.

In the offline Delete Archive Process, Archive compares Archive File data with data in one or more DB2 UNLOAD-compatible files to create a new file that excludes the archived data marked “delete after archive.” Archive uses matching keys as the basis of comparison for the two files. The new file, minus the deleted data, can then be loaded into the database.

The offline Delete Archive Process requires the following steps:

1. Use an online or batch Archive Process to create an Archive File containing one or more tables designated for a deferred delete after archive.
2. Unload data from the database using DSNTIAUL, REORG, REORG UNLOAD, or an unload program. Each unload file must contain all columns of the table.

Note: Do not unload multiple tables in one process, using DSNTIAUL.

1. Run the offline delete utility with one or more Archive Files and the unloaded data as input. Because an unload file typically contains data from a single tablespace, which is often limited to data from a single table, multiple unload files can be used as input to this offline process. The output is in two parts:

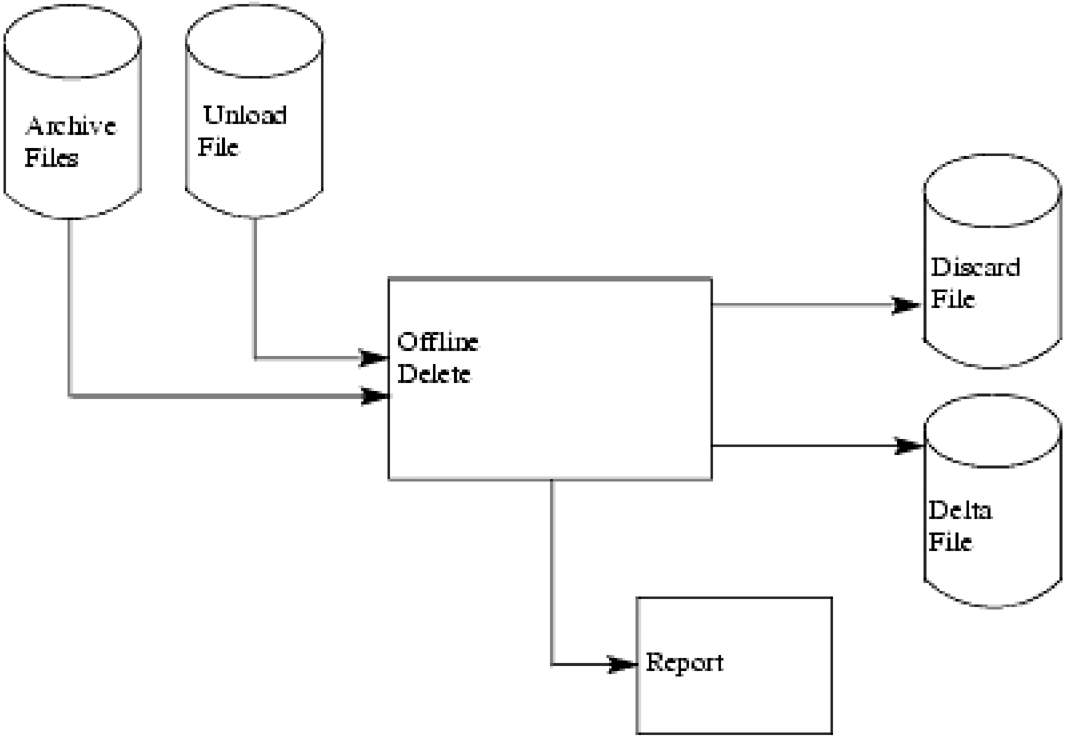
   Note:

   - The Discard File contains only unloaded data that matches data in the Archive Files that is designated for delete after archive.
   - The Delta File contains all other unloaded data.

The format of these output files is the same as the format of the unloaded data.

2. Load the Delta File to the database using DB2 LOAD or LOADPLUS. You must include the LOAD utility REPLACE option to delete all rows in the tablespace before copying the Delta rows into the database. If it becomes necessary, the Discard File can also be reloaded to the database.

The following diagram illustrates the offline Delete Archive Process:



The JCL needed to execute the offline delete utility is:

```
//          EXEC PGM=FOPMAIN,REGION=0M,
//          PARM='CON UTILITY subsys planname sqlid userid'
//*
//PSDFDFLT  DD DSN=your.loadlib(FOPMDFLT)  DEFAULT SITE OPTIONS
//              DISP=SHR
//SYSPRINT  DD SYSOUT=*                    INTERNAL ERRORS / MESSAGES
//SYSTERM   DD SYSOUT=*                    INTERNAL ERRORS / MESSAGES
//PSTRACE   DD SYSOUT=*                    TRACE OUTPUT
//PSDFASUM  DD SYSOUT=*                    UTILITY SUMMARY LISTING FILE
//PSDFADIR  DD SYSOUT=*                    DIRECTORY LISTING FILE
//PSDFAREP  DD SYSOUT=*                    REPORT OUTPUT
//PSDFRPRT  DD SYSOUT=*                    DELETE REPORT OUTPUT
//SYSRECnn  DD DSN=input.file.name         INPUT UNLOAD-COMPATIBLE FILES
//              DISP=SHR
//SYSNEWnn  DD DSN=output.file.name        OUTPUT FILES
//              DISP=OLD
//SYSDSCnn  DD DSN=discard.file.name       DISCARD FILES
//              DISP=OLD
//SYSIN     DD *
               control statements
//*
```

A sample of this JCL is provided in the sample library as member FOPJOBUT. Unique data set names are required for SYSREC*nn*, SYSNEW*nn*, and SYSDSC*nn*; the offline delete utility will not overwrite the input file.

OFFLINE_DELETE Statement

Use the OFFLINE_DELETE statement to compare data in one or more Archive Files with data in one or more DB2 UNLOAD-compatible files. This creates new UNLOAD-compatible files that exclude the archived data marked “delete after archive.” The new files, minus the deleted data, can then be loaded into the database.

Executing the OFFLINE_DELETE statement also generates a report of the Delete Process in the Delete Report listing file. Information in the report includes the input data set name, Discard File name, and names of any tables not processed.

```
OFFLINE_DELETE
NAME ( arc.file.name1 [, arc.file.name2, ... ] )
UNLD ( [ cid. ] tblname1, unld-ddn, discard-ddn, outunld-ddn,
      format, scan [, VIEWDELETE ] )
[ UNLD ( [ cid. ] tblname2, unld-ddn, discard-ddn, outunld-ddn,
      format, scan [, VIEWDELETE ] ) ]
```

Note: ARCHIVE_DELETE is a synonym for the OFFLINE_DELETE statement.

The keywords and operands in the OFFLINE_DELETE statement are:

NAME

The name of one or more Archive Files with data to be deleted. Multiple values can be specified in any order, but must be enclosed in parentheses and separated by commas. (Note that if you specify a single value, parentheses are optional.) This keyword and its operand must be included in the OFFLINE_DELETE statement.

arc.file.name

Specify the fully qualified name of a valid Archive File data set as an explicit value or a pattern, using DB2 LIKE syntax. An Archive entry for the file is not required.

UNLD

Processing for tables in the Archive File. This keyword and the following operands are required for each table. The operands are positional and must be specified in the following order:

cid.tblname

Specify the name of a table marked for DAA in the designated Archive File and prefix with Creator ID, if different from the default Creator ID for the Archive File. An error condition occurs if the table is not designated for DAA.

unld-ddn

DDNAME for the corresponding UNLOAD-compatible file used as input to the offline Delete Archive Process.

discard-ddn

DDNAME for the output Discard File.

outunld-ddn

DDNAME for the output Delta File.

format Format of the UNLOAD-compatible input, Delta, and Discard files. Specify the format of the input file:

BMC UNLOAD PLUS format.

DSNTIAUL

DSNTIAUL format, including Auto-Unload, Auto-Online Unload, and IBM High Performance Unload.

UNLOADONLY

IBM REORG UNLOAD ONLY

UNLOADEXT
IBM REORG UNLOAD EXTERNAL

UNLOAD
IBM REORG UNLOAD CONTINUE or UNLOAD PAUSE format.

scan The method used to scan the Unload and Archives Files for key matches. Specify one of the following:

MATCHSCAN
If a key match exists, scan the remainder of the row to verify that no values have changed. If a change has occurred, the row is retained.

MATCHCRC
If a key match exists, compare a generated CRC value instead of the remainder of the row. If a change has occurred, the row is retained.

MATCHNONE
The user must verify that no changes occurred since the Archive Process. If a key match exists, the row is written to the Discard File. Duplicate keys are not supported and will result in a duplicate key error.

VIEWDELETE
If processing a view, use this keyword to match only the columns that are present. If you omit this keyword when processing a view or if the view applies to more than one table, an error condition occurs.

Note: The entire row is not matched; values in columns other not represented in the view may have been changed.

Example

The following is an example of OFFLINE_DELETE statement usage:

The DETAILS and ORDERS tables, in the input UNLOAD-compatible files associated with the DDNAMES SYSREC00 and SYSREC01, are designated DAA in the Archive File FOPDEMO.ARCHIVE.FILE. The DDNAMES SYSDSC00 and SYSDSC01 designate the Discard Files, and the DDNAMES SYSNEW00 and SYSNEW01 designate the Delta Files. A matchscan is performed. To purge the designated data, specify:

```
OFFLINE_DELETE
   NAME FOPDEMO.ARCHIVE.FILE
   UNLD (DETAILS, SYSREC00, SYSDSC00, SYSNEW00,
      UNLOAD, MATCHSCAN)
   UNLD (ORDERS, SYSREC01, SYSDSC01, SYSNEW01,
      UNLOAD, MATCHSCAN)
```

COMPARE

Use a COMPARE statement to compare two sets of related data. The data to be compared must be specified in a Compare Definition, created online in the Optim Directory.

You can compare a set of rows that resides in multiple database tables or rows extracted previously and stored on disk in an Extract or Archive File. Results of a comparison are stored in a Compare File or can be used to create a report. Most options that may be specified using the online Compare panel are also available using the COMPARE batch processing statement.

Note: An extract file on tape cannot be used in a compare process.

Refer to the *Compare User Manual* or *Compare for IMS, VSAM or Sequential File Data* for details.

```
COMPARE
  COMPARE_FILE ( File Allocation Parameters )
  { COMPARE_DEFINITION_DEFINE (parameters); | COMPARE_DEFINITION group.user.name
     [ SOURCE_FILE_OVERRIDE dsname ]
     [ SOURCE_CID_OVERRIDE_{ 1 | 2 } { cid | % } ]
     [ MAX_EXTRACT_ROWS nn ]
     [ UNLOAD_UTILITY { BMC | IBM }
        [ IMAGE_COPY
            [ MODE { L | A | B | D } ]
            [ DATE yyyy-mm-dd ]
            [ TIME hh.mm.ss ]
            [ DSNAME dsname ] ) ]
        [ PARTITIONS_n ( part1,...partn )
              [ OTHER_PARTITIONS_SAME { YES | NO }] ]
      [ MATCH_KEY ( cid.tablename,oldkeycol,newkeycol  ) ]
     [ REPORT_MODE { YES | NO | ONLY }
        REPORT_FILE ( File Allocation Parameters )
        [ REPORT_LINES nn ]
        [ REPORT_TABLE [ cid. ] tablename]
        [ REPORT_TYPE { SUMMARY | { DETAIL | BOTH  } } ]
        [ REPORT_FORMAT { COLUMN | SIDE_DIFF | SIDE_ALL | EXTERNAL } ]
        [ ROW_DISPLAY ( DIRECT | RELATED | UNMATCHED | ORPHAN | DUPLICATE | UNUSED ) ]
        [ WRAP_LINES { EXTEND | WRAP } ] ]
        [ TABLE (cid.tablename,dsname)]
            [ STRING_DELIMITER 'c']
            [ ESCAPE CHARACTER 'c']
            [ FIELD_DELIMITER ',']
            [ GENERATE_HEADER {YES | NO}]
                  [ BEGIN_LABEL label ]
                  [ END_LABEL label ]
                  [ HEADER_DELIMITER ',']
                  [ MATCH_KEY_LABEL 'c' ]
                  [ REL_KEY_LABEL 'c' ]
                  [ USE_COLUMN_LABELS {YES | NO}]
               [ USE_DATA_LABELS {YES | NO}
               [ CHANGE_DATA_LABEL 'c' ]
               [ EQUAL_DATA_LABEL 'c'] }
```

Compare File parameters

COMPARE_FILE

Name of the Compare File that is to contain the results of the comparison. Use the parameters in File Allocation Parameters to name and allocate the Compare File.

If you use the REPORT_MODE ONLY parameter, you need only specify the COMPARE_FILE data set name.

COMPARE_DEFINITION_DEFINE

The Optim online process generates this parameter when it creates a COMPARE job for batch execution.

When you create a COMPARE job outside of the Optim online process, the best practice is to use the COMPARE_DEFINITION parameter to refer to a named Compare Definition in the Optim Directory.

Place the COMPARE_DEFINITION_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See COMPARE_DEFINITION_DEFINE Parameters for the allowable parameters.)

Compare Definition parameters

Use the following parameters to identify the Compare Definition and override or augment various parameters in it.

COMPARE_DEFINITION

The name of the Compare Definition. COMPARE_DEFINITION must be included in the

COMPARE statement and must precede any parameters that override parameters in the Compare Definition (e.g., SOURCE_FILE_OVERRIDE_1).

group.user.name
The three-part Compare Definition name.

SOURCE_FILE_OVERRIDE
Override for the name of a file specified as a source in the Compare Definition. To override the Source 1 specification, use SOURCE_FILE_OVERRIDE_1. Use SOURCE_FILE_OVERRIDE_2 to override the Source 2 specification.

dsname
The fully qualified name of the Archive File, Extract File or Access Definition.

SOURCE_CID_OVERRIDE_{ 1 | 2 }
Override for the default creator ID for a source in the Compare Definition. To override the Source 1 specification, use SOURCE_CID_OVERRIDE_1. Use SOURCE_CID_OVERRIDE_2 to override the Source 2 specification.

cid Creator ID to override the default Creator ID for the source.

If the source is an Extract File, you can use the wildcard (%) to use the default Creator ID defined in the Extract File.

This Creator ID applies to all tables that were defined with the initial default Creator ID in the Compare Definition. Tables in the Compare Definition not defined with the initial default Creator ID will not be altered by the specified Creator ID override. This override also changes the names of the tables in the relationships processed by Compare to use the override default Creator ID. If a relationship is not found for the updated table name, an error occurs when the compare is performed. If the source is an extract file, and a table initially defined in the Compare definition with default Creator ID is not in the extract file, the table will be skipped, a warning message will be produced, but the Compare job will continue.

MAX_EXTRACT_ROWS
The maximum number of rows that can be extracted from the source table. If this value is exceeded, the Compare process terminates. The default is the site option value for Maximum Extract Rows. The Compare Definition must specify a DB2 table or an Access Definition as a source. Use MAX_EXTRACT_ROWS_1 for Source 1 and MAX_EXTRACT_ROWS_2 for Source 2.

nn The maximum number of rows to extract during the Compare process.

Unload Utility parameters

UNLOAD_UTILITY
The name of the unload utility used to retrieve the data. If omitted, DB2 is used to retrieve the data. This parameter is required to extract data from an image copy.

Use these parameters only for a Compare Definition that specifies a DB2 table or an Access Definition as a source. Use UNLOAD_UTILITY_1 for Source 1 and UNLOAD_UTILITY_2 for Source 2.

BMC Use the UNLOAD PLUS utility.

IBM Use the High Performance Unload utility

Note: You must add the appropriate utility DD statements. Refer to the utility documentation for the required DD statements.

IMAGE_COPY
The image copy file(s) to be used as input for a source. If omitted, the DB2 database is

used as input. IMAGE_COPY_1 is allowed only if UNLOAD_UTILITY_1 is used; IMAGE_COPY_2 is allowed only if UNLOAD_UTILITY_2 is used.

Note: To extract or compare data from DB2 image copy data sets in multiple partitions of the same tablespace stored on the same tape volume, you must manually edit the JCL to allocate the data sets. Multiple image copy data sets cataloged on the same tape volume can not be allocated using dynamic allocation. This is a z/OS limitation. If you attempt to use dynamic allocation, the extract or compare process fails with a dynamic allocation error. Refer to Chapter 5, “File Allocation Parameters,” on page 139 for details.

MODE

The image copy file to use:

L Use the latest file.

A Use the first file created on or after the specified DATE and TIME.

B Use the first file created on or before the specified DATE and TIME.

D Use the file named in DSNAME.

DATE The date criteria for the image copy file in this format: *yyyy-mm-dd*.

TIME The time criteria for the image copy file in this format: *hh.mm.ss*. If omitted, a value of 00.00.01 is assumed.

DSNAME

The fully qualified name of the image copy file.

PARTITIONS

The partitions to be processed by an unload utility. Valid only if the Start Table for the source is partitioned. Use PARTITIONS_1 for Source 1 and PARTITIONS_2 for Source 2. Omit this parameter to process all partitions.

(part1,...partn)

The partition numbers. Partitions that are not specified are not compared.

OTHER_PARTITIONS_SAME

Use this operand to indicate whether partitions specified in the PARTITIONS parameter for the Start Table should be applied to all tables in the Compare Definition.

YES The partitions specified in the PARTITIONS parameter for the Start Table will be applied to all tables in the Compare Definition.

NO The partitions specified in the PARTITIONS operand will apply only to the Start Table. All partitions for all other tables in the Compare Definition will be processed. This is the default.

MATCH_KEY

Replace a key column in the Match Key for the specified table.

(cid.tablename,oldkeycol,newkeycol)

The fully qualified tablename, old key column name, new key column name.

Report parameters

REPORT_MODE

Specify whether a report is produced.

YES Produce a report as part of the Compare process.

NO Do not produce a report (default).

ONLY Produce a report from a Compare file created previously and specified in the COMPARE_FILE parameter.

REPORT_FILE
Name of the file for the report. Use the parameters in File Allocation Parameters to name and allocate the file. This parameter is required if REPORT_MODE is YES or ONLY.

REPORT_LINES
The number of lines per page for the report.

nn Number of lines per page. Valid values are 0-99. The default is 57. Specify 0 to suppress all page breaks and title lines.

REPORT_TABLE
The name of a table for which the report is generated when multiple tables have been compared. Omit this parameter to include all tables.

[*cid* **.]***tablename*
The name of the table for which the report is generated. If the creator ID is omitted, all tables with the specified name are included, regardless of the creator ID.

REPORT_TYPE
The type of report to be produced.

SUMMARY
Produce a report containing summary information. This is the default.

{ DETAIL | BOTH }
Specify `DETAIL` or `BOTH` to produce a report containing both summary and detailed information.

ROW_DISPLAY
Rows that appear on a detailed report. If you omit this parameter, all rows are included. Operands must be enclosed in parentheses, separated by commas, and may appear in any order.

DIRECT
Include rows with the same Match Key value that have different values in one or more other columns.

RELATED
Include rows marked as having related changes.

UNMATCHED
Include rows with match key value that does not match any row from the other source.

ORPHAN
Include rows that do not have a parent.

DUPLICATE
Include rows that have duplicate match key values.

UNUSED
Include unused columns in the report.

WRAP_LINES
Option for rows that exceed the line width of the report file for a detailed report.

WRAP
Wrap the report data (default).

EXTEND
Increase the record length of the report file to fit the row.

REPORT_FORMAT
The format of a detailed report.

COLUMN
Report is in columnar format (default).

SIDE_ALL
Report is in sidelabels format and includes all columns.

SIDE_DIFF
Report is in sidelabels format and includes only changed columns and Match Key columns.

EXTERNAL
Report is in CSV (Comma Separated Values) format.

The following parameters are used only when REPORT_FORMAT=EXTERNAL is specified.

TABLE
Write the output for a specific table to an external file other than the file specified in the REPORT_FILE parameter. All file allocation parameters for the REPORT_FILE parameter, except dsname, apply to this file. You can use multiple TABLE parameters, provided each refers to a different table.

(*cid.tablename,dsname***)**
The fully qualified table name and data set name, enclosed in parentheses and separated by a comma.

STRING_DELIMITER
Specify a single character used to separate character literal values.

ESCAPE_CHARACTER
Specify a single character used to generate the value of a character normally used as a delimiter.

FIELD_DELIMITER
Specify a single character used to separate values in a row. A comma (“,”) is the default.

GENERATE_HEADER
Generate headers using the column names.

Y Generate headers

N Do not generate headers. This is the default.

BEGIN_LABEL
Place a label before the first column in the table. This parameter applies only when GENERATE_HEADER=Y. Specify “$table” to include the table name in the label.

END_LABEL
Specify a label to be placed after the last column in table. This parameter applies only when GENERATE_HEADER=Y. There is no default.

HEADER_DELIMITER
Character used to separate column headings. This parameter applies only when GENERATE_HEADER=Y. A comma (“,”) is the default.

MATCH_KEY_LABEL
Specify a single character to be placed before each column that was used as part of the Match Key between Source 1 and Source 2 during compare processing.. This parameter applies only when GENERATE_HEADER=Y.

REL_KEY_LABEL
Specify a single character to be placed before each column that was part of a

relationship when tables were joined during compare processing. This parameter applies only when GENERATE_HEADER=Y.

USE_COLUMN_LABELS
Indicator for using DB2 column labels or column names in the header, if a header is generated. Specify:

Y Use column labels.

N Do not use column labels. Use column names. This is the default.

USE_DATA_LABELS
Specify whether to place a single character before the data of each column to indicate if the column data in Source 1 and Source 2 is equal or unequal.

Y Use data labels.

N Do not use data labels. This is the default.

CHANGE_DATA_LABEL
Specify a single character to be placed before column data that differs between Source 1 and Source 2. This parameter applies only when USE_DATA_LABELS=Y.

EQUAL_DATA_LABEL
Specify a single character to be placed before column data that is equal between Source 1 and Source 2. This parameter applies only when USE_DATA_LABELS=Y.

Example

This example compares two sets of data defined by the Compare Definition PSTUSER.INV.CDQ4. The Extract File PSTUSER.ITEMS is used in place of the Extract File specified for Source 1 in the Compare Definition:

```
COMPARE_FILE  (DSNAME PSTUSER.INVCOMP30)
   COMPARE_DEFINITION  PSTUSER.INV.CDQ4
   SOURCE_FILE_OVERRIDE_1 PSTUSER.ITEMS
```

COMPARE Override Parameters

When you submit the job, you can specify overrides to the Source 1, Source 2, or both data set names, and the default Creator ID for the source tables. Also, if an unload program is used to extract data for Source 1, Source 2, or both, you can override the Image Copy data set name, date, and time parameters for each source.

Use the PSDFOVRD DD statement in the JCL to provide the desired overrides.

COMPARE_DSN_SRC1

To specify a new data set name for the Source 1 file, specify:

```
COMPARE_DSN_SRC1     data.set.name
```

This override allows you to use one set of saved JCL to compare files regardless of the names.

COMPARE_DSN_SRC2

To specify a new data set name for the Source 2 file, specify:

```
COMPARE_DSN_SRC2     data.set.name
```

This override allows you to use one set of saved JCL to compare files regardless of the names.

SOURCE_CID_OVERRIDE

To override the default Creator ID for the tables in Source 1 or Source 2 of the Compare Definition, specify:

```
SOURCE_CID_OVERRIDE_{ 1 | 2 } {cid | % }
```

To override the Source 1 specification, use SOURCE_CID_OVERRIDE_1. Use SOURCE_CID_OVERRIDE_2 to override the Source 2 specification.

cid Creator ID to override the default Creator ID for the source.

If the source is an Extract File, you can use the wildcard (%) to use the default Creator ID defined in the Extract File.

This Creator ID applies to all tables that were defined with the initial default Creator ID in the Compare Definition. Tables in the Compare Definition not defined with the initial default Creator ID will not be altered by the specified Creator ID override. This override also changes the names of the tables in the relationships processed by Compare to use the override default Creator ID. If a relationship is not found for the updated table name, an error occurs when the compare is performed. If the source is an extract file, and a table initially defined in the Compare definition with default Creator ID is not in the extract file, the table will be skipped, a warning message will be produced, but the Compare job will continue.

UNL_IMAGECOPY_DSN

To override the Image Copy DSN parameter for an unload program used to extract data for Source 1, specify:

```
UNL_IMAGECOPY_DSN     image.file.dsn
```

UNL_IMAGECOPY_DATE

To override the Image Copy Date parameter for an unload program used to extract data for Source 1, specify:

```
UNL_IMAGECOPY_DATE     yyyy-mm-dd
```

UNL_IMAGECOPY_TIME

To override the Image Copy Time parameter for an unload program used to extract data for Source 1, specify:

```
UNL_IMAGECOPY_TIME     hh.mm.ss
```

UNL_IMAGECOPY_SELECT

To override the Image Copy Criteria parameter for an unload program used to extract data for Source 1, specify:

```
UNL_IMAGECOPY_SELECT     { A | B | L | S }
```

A First Image Copy file created on or after the specified Date and Time.

B First Image Copy File created on or before the specified Date and Time.

L Latest Image Copy file. Any Date and Time values are ignored.

S Image Copy file. The name is provided as the Image Copy DSN parameter.

UNL_IMAGECOPY_DSN2

To override the Image Copy DSN parameter for an unload program used to extract data for Source 2, specify:

```
UNL_IMAGECOPY_DSN2     image.file.dsn
```

UNL_IMAGECOPY_DATE2

To override the Image Copy Date parameter for an unload program used to extract data for Source 2, specify:

```
UNL_IMAGECOPY_DATE2     yyyy-mm-dd
```

UNL_IMAGECOPY_TIME2

To override the Image Copy Time parameter for an unload program used to extract data for Source 2, specify:

```
UNL_IMAGECOPY_TIME2     hh.mm.ss
```

UNL_IMAGECOPY_SELECT2

To override the Image Copy Criteria parameter for an unload program used to extract data for Source 2, specify:

```
UNL_IMAGECOPY_SELECT2     { A | B | L | S }
```

A First Image Copy file created on or after the specified Date and Time.

B First Image Copy File created on or before the specified Date and Time.

L Latest Image Copy file. Any Date and Time values are ignored.

S Image Copy file. The name is provided as the Image Copy DSN parameter.

CONVERT

Use the CONVERT statement to transform data in an extract or archive file. You can use this process to mask sensitive data, or to convert data to a CSV (comma separated values) format. The converted file can be used to create reports, while the original file can be retained for audit or restore purposes. Most options that may be specified using the online CONVERT panel are also available using the batch CONVERT processing statement.

Note: When you convert a file that is stored on tape, the converted file will be written to disk.

```
CONVERT
  INPUT_FILE { (USEDD) | explicitfilename }
  CONVERTED_FILE (  File Allocation Parameters )
  CONTROL_FILE  ( File Allocation Parameters )
  { TABLE_MAP mapid.name | TABLE_MAP_DEFINE (parameters) ; }
  [ DISCARD_COUNT n ]
  [ REPORT_LEVEL { DETAIL | SUMMARY } ]
  [ SORT_ROWS {YES | NO } ]
  [ AGING (parameters) ]
  [ EXTERNAL_FILE ( File Allocation Parameters )
     [ TABLE (creator.tablename,dsname) ]
     [ FIELD_DELIMITER 'c' ]
     [ STRING_DELIMITER 'c' ]
     [ ESCAPE_CHARACTER 'c' ]
     [ GENERATE_HEADER{ YES | NO }
        [ BEGIN_LABEL string ]
        [ END_LABEL string ]
        [ HEADER_DELIMITER 'c' ]
        [ USE_COLUMN_LABELS { YES | NO } ] ] ]
```

INPUT_FILE
The fully qualified name of the Archive or Extract File to be converted. INPUT_FILE must be included in the CONVERT statement.

INPUT_FILE values must be enclosed in parentheses. DSNAME is required; other keywords and operands are optional, depending on site requirements.

USEDD
Use the file name specified in the PSDFEXTR control card statement.

explicitfilename
The fully qualified name of the Archive or Extract File

CONVERTED_FILE
The converted Extract or Archive File. See File Allocation Parameters to name and allocate the converted file. This keyword is required when

- INPUT_FILE is an Extract File and you want to write the converted data to a new Extract File
- INPUT_FILE is an Archive File and you want to write the converted data to a new Archive File

Omit this keyword when:

- INPUT_FILE is an Extract or Archive file and you want to write the converted data to an external file in CSV format
- INPUT_FILE is an Extract File, and you want the converted data to replace the contents of the input Extract File

CONTROL_FILE
The Control File. This keyword is required. Use the keywords in File Allocation Parameters to name and allocate the control file.

TABLE_MAP
The name of the table map to be used. TABLE_MAP is required if the converted file is an Archive or Extract File. Omit this keyword if the converted file is an external file in CSV format.

mapid.name
The fully qualified name of a new Table Map.

TABLE_MAP_DEFINE
The Optim online process generates this keyword when it creates a CONVERT job for batch execution.

When you create a CONVERT job outside of the Optim online process, the best practice is to use the TABLE_MAP keyword to refer to a named Table Map in the Optim Directory.

Place the TABLE_MAP_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See TABLE_MAP_DEFINE Parameters for the allowable parameters.)

DISCARD_COUNT
The maximum number of discarded rows for the CONVERT process. If the maximum value is exceeded, the CONVERT process terminates. Omit this keyword to allow an unlimited number of rows to be discarded.

n Number in the range 1 - 4,294,967,295.

REPORT_LEVEL
The level of detail provided in the Process Report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

SORT_ROWS
Option to sort rows for destination tables with a cluster index. Specify:

YES Sort rows

NO Do not sort rows (default).

AGING
This keyword indicates that date values in the source columns are to be aged. It provides parameters to be used in the aging process. See AGING Keyword Parameters for detailed information on using this keyword.

EXTERNAL_FILE
Option to convert to an external file in CSV format. (The output for each table in the report will be directed to this file unless a TABLE keyword is supplied to direct the output for a specific table to a different file.) Omit the CONVERTED_FILE keyword if you use this keyword. See File Allocation Parameters for keywords to name and allocate the external file.

Use the following keywords to control options for the external file. These keywords apply only if you use the EXTERNAL_FILE keyword.

TABLE
Write the converted output for a specific table to an external file other than the file specified in the EXTERNAL_FILE keyword. All file allocation parameters for the EXTERNAL_FILE keyword, except dsname, apply to this file. You can use multiple TABLE keywords, provided each refers to a different table.

(*cid.tblname,dsname***)**
The fully qualified table name and data set name, enclosed in parentheses and separated by a comma.

FIELD_DELIMITER
The character used to separate values in a row. A comma is the default.

'c' The field delimiter character, enclosed in single (‘ ') or double quotes (“ ”).

STRING_DELIMITER
The character used to separate character literal values. A blank is the default.

‘c' The string delimiter character, enclosed in single (‘ ') or double quotes (“ ”).

ESCAPE_CHARACTER
The character used to generate the value of a character normally used as a delimiter.

‘c' The escape delimiter character, enclosed in single (‘ ') or double quotes (“ ”).

GENERATE_HEADER
Option to generate a header for the external file.

YES Generate a header.

NO Do not generate a header (default).

The following keywords are valid only if GENERATE_HEADER is YES.

BEGIN_LABEL
Option to place a label before the first column in the table.

string The character string (up to 10 alphanumeric characters) to use as a label. To include the table name, specify $*table*.

Note: If you enclose the string in quotes, the quotes are considered part of the string and included in the header.

END_LABEL
Option to place a label after the last column in the table.

string The character string (up to 10 alphanumeric characters) to use as a label.

Note: If you enclose the string in quotes, the quotes are considered part of the string and included in the header.

HEADER_DELIMITER

The character used to separate column headings. A comma is the default.

‘c’ The header delimiter character, enclosed in single (‘ ') or double quotes (“ ”).

USE_COLUMN_LABELS

Indicate whether to use column labels in the header.

YES Use column labels in the header.

NO Do not use column labels in the header (default).

Examples

The following are examples of CONVERT batch statement usage:

1. To convert an Archive File, write the converted data to a new Archive File, and create a summary process report, specify:

```
CONVERT INPUT_FILE PSTUSER.ARCSALES
   CONVERTED_FILE (DSNAME PSTHR.SAL)
   CONTROL_FILE (DSNAME PSTUSER.CVCON)
   TABLE_MAP PSTUSER.MAP11
   REPORT_LEVEL SUMMARY
```

2. To convert an Extract File to CSV format, allocate 60 tracks of space for the output file, and generate a header that has the table name as its label, specify:

```
CONVERT INPUT_FILE PSTUSER.NESALE
   CONTROL_FILE (PSTUSER.SALECTRL)
   EXTERNAL_FILE (DSNAME  PSTUSER.SALECSV
                  PRIMARY  60T)
   GENERATE_HEADER YES
   BEGIN_LABEL $table
```

DEFERRED_DELETE

Use a DEFERRED_DELETE statement to delete archived data from the database when the delete was deferred during the Archive Process. This statement uses DB2 SQL statements to perform the delete. You can use keywords to provide parameters similar to those for the online Archive Delete Process.

Note: If you are deleting large amounts of data, it may be more efficient to use the OFFLINE_DELETE statement. This statement deletes archived data, but avoids the overhead of DB2 logging. (For more information, see “Offline Delete Archive Process” on page 72.)

```
DEFERRED_DELETE
  ARCHIVE_NAME { (USEDD) | explicitfilename
  CONTROL_FILE ( File Allocation Parameters )
  [ DELETE_LOCK { YES | NO } ]
  [ DELETE_COMMIT_ROWS n ]
  [ DELETE_COMMIT_MINUTES n ]
  [ DELETE_DISCARD n ]
  [ ACTION_IF_NO_ROWS { WARNING | ERROR }
  [ RESTART { YES | NO } ]
  [ COMPARE_ROW { YES | NO } ]
  [ DAA_TABLE ( [ cid. ] tablename, { YES | NO } ) ]
  [ DEFAULT_KEY_LIMIT n ]
  [ ACCESS_METHOD ( [ cid. ] tablename, { K | S | E },key limit ) ]
  [ REPORT_LEVEL { DETAIL | SUMMARY } ]
  [ NOT_FOUND_RC4 { YES | NO } ]
  [ EMPTY_TABLE_FOUND_RC4 { YES | NO } ]
  [ DELETE_0_ROWS_RC4 { YES | NO } ]
  [ DELETE_FAILED_ROWS_RC4 { YES | NO } ]
```

Archive File Keyword

Use the following keyword to name the Archive File.

ARCHIVE_NAME

The name of the Archive File to be processed. ARCHIVE_NAME must be included in the DEFERRED_DELETE statement.

USEDD

Use the file name specified in the PSDFEXTR control card statement.

explicitfilename

The fully qualified name of the Archive File

Control File Keywords

CONTROL_FILE

Name of the Control File and its allocation parameters. This keyword is required to execute the Delete Process. Use the keywords shown in File Allocation Parameters to name and allocate the Control File.

Processing Keywords

Use the following keywords to specify processing options for the Archive Delete Process.

DELETE_LOCK

Indicate whether to lock tables during the Delete Process. Locking tables ensures other database activity does not interfere with the Delete Process. However, it will prevent other users from accessing the table.

YES Lock tables. (This setting causes an error if site options prevent a user from locking tables.)

NO Do not lock tables (default).

DELETE_COMMIT_ROWS

The frequency of commits in rows for the Delete Process. The commit points affect the starting point in case of a Restart. Frequent commits keep page locks to a minimum. If you omit both DELETE_COMMIT_ROWS and DELETE_COMMIT_MINUTES, the site limit is used.

n Commit each *n* number of rows where *n* is a value from 1 through the Site Options Commit Frequency Rate, inclusive.

Note: If DELETE_LOCK is set to YES, DELETE_COMMIT_ROWS and DELETE_COMMIT_MINUTES are ignored. A commit is performed as processing for each table is completed.

DELETE_COMMIT_MINUTES

The frequency of commits in minutes for the Delete Process. If you omit DELETE_COMMIT_MINUTES, DELETE_COMMIT_ROWS determines the commit frequency.

n Commit every *n* minutes, where *n* is 1 through 1440.

DELETE_DISCARD

The maximum number of rows that can be discarded when deleting rows, before terminating the process. If the limit is met, the process is terminated. You can use Restart to begin the Delete Process again at the termination point.

n Specify a value in the range 1 to 4,294,967,295.

blank No limit on discarded rows.

ACTION_IF_NO_ROWS

The action to be taken if no rows are deleted from the database. That is, should the utility generate a warning or an error condition?

WARNING
Treat no rows being deleted as a warning condition. The utility issues a warning message and continues processing (default).

ERROR
Treat no rows being deleted as an error condition.

RESTART
Indicate whether to restart or retry the delete portion of an Archive Process if a processing failure occurs. Optim automatically determines whether to perform a restart or a retry.

YES Restart or retry the delete, and, optionally, respecify any DELETE keywords.

NO Do not attempt to restart or retry the delete (default).

Note: You can also restart or retry the process by adding the RESTART operand to the PARM field on the batch EXEC statement. Use a blank to separate the RESTART operand from the previous operand in the PARM field.

COMPARE_ROW
Indicate whether the Delete Process compares rows of data in the Archive File with rows in the database prior to deletion.

YES Rows are deleted from the database only if they exactly match rows in the Archive File. Rows that do not exactly match are discarded and noted in the Control File. (This setting causes an error if the **Compare Row Contents** site option is not set to USER.)

NO Row comparison is not performed (default). This may improve performance significantly; however, you risk losing any updates to the data in the database since the Extract Process was performed.

DAA_TABLE
Specifies whether a table in the Access Definition is to be deleted. The default value for each table is taken from the value specified in the ARCHIVE statement that created the Archive File or, if not specified there, from the Access Definition itself. An existing value of NO may not be changed to YES. Multiple keywords are permitted as long as the table name is not repeated.

cid.tablename
Name of the table to be deleted.

YES Delete the table.

NO Do not delete the table.

DEFAULT_KEY_LIMIT
The default maximum number of keys used at one time to process a table with key lookup. Applies only if the Access Definition does not specify a limit.

n 1 - 100 (default is 1)

The following conditions must be true to process multiple keys at one time when deleting rows:

- An index on the primary key is defined for the table.
- COMPARE_ROW keyword is NO.
- Row-level Archive Actions are not defined for the Delete Process (e.g., Before Delete of Row).
- The table is not a parent in a DBMS relationship.

ACCESS_METHOD
Indicate how to access the rows in a DB2 table. You may specify this keyword once for

each table to be processed. If you omit it for a table, Optim determines how to access the rows in that table. You must specify the parameters within parentheses, separated by commas, and in the following order.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

cid.tablename
The table name.

K Use key lookup.

S Use table scan.

E Optim determines the access method.

key limit
Maximum number of keys to be specified in an SQL statement when the access method is K. Specify a number between 1 and 100. You may omit this parameter if the access method is S.

REPORT_LEVEL
The level of detail provided in the Process Report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

NOT_FOUND_RC4
Indicate the action taken and return code (RC) setting when no rows have been found during Delete batch processing.

Note: If the source file has no rows, Optim ignores this parameter and issues a warning and RC=8 or RC=4, depending on the specification for ACTION_IF_NO_ROWS.

YES A warning message will be issued and the program will end with RC=4 (default).

NO The program will end with RC=0.

EMPTY_TABLE_FOUND_RC4
Indicate the action taken and return code (RC) setting when empty tables have been detected during Delete batch processing.

Note: If the source file has no rows, Optim ignores this parameter and issues a warning and RC=8 or RC=4, depending on the specification for ACTION_IF_NO_ROWS.

YES Empty tables will be processed, a warning message will be issued, and processing will end with RC=4.

NO Empty tables will be ignored and processing will end with RC=0 (default).

DELETE_0_ROWS_RC4
Indicate the action taken and the return code (RC) setting during Delete batch processing when a table has rows in an Archive File or Extract File but no rows have been deleted. Possible reasons include the rows cannot be deleted because there are no matching rows in DB2 tables or a condition existed that prevented it (for example, a violation of Referential Integrity would occur).

YES A warning message will be issued and processing will end with RC=4.

NO Processing will end with RC=0 (default).

DELETE_FAILED_ROWS_RC4

Indicate the action taken and return code (RC) setting when some rows of a table have failed to be deleted because a condition existed that prevented it (for example, a violation of Referential Integrity would occur).

YES A warning message will be issued and processing will end with RC=4.

NO Processing will end with RC=0 (default).

Example

Specify the following to delete archived data using an Archive File named PSTUSER.ARCHIVE.CUST, and a Control File named PSTUSER.CONTROL. This example also locks tables and specifies 5000 for the maximum number of discarded rows.

```
DEFERRED_DELETE
   ARCHIVE_NAME PSTUSER.ARCHIVE.CUST
   CONTROL_FILE (DSNAME PSTUSER.CONTROL)
   DELETE_LOCK YES
   DELETE_DISCARD 5000
```

DEFERRED_DELETE Override Keywords

You can provide overrides to specifications for DEFERRED_DELETE to replace the default Creator ID, selection criteria, and SQL WHERE clause in the Access Definition. This is especially useful when performing several DEFERRED_DELETE Processes that vary only by Creator ID for the tables or by selection criteria.

Use the PSDFOVRD DD statement to provide the desired overrides.

ARCHIVE_DSN

To specify a new data set name for the Archive File, specify:

```
ARCHIVE_DSN     data.set.name
```

This override allows you to use one set of saved JCL to delete Archive Files with different names.

DEFCID

To override the default Creator ID in the Access Definition, specify:

```
DEFCID     cid
```

cid The default Creator ID. This Creator ID applies only to tables that are not explicitly qualified in the Access Definition.

This override also affects the names of the tables in the relationships on the Relationship Usage list. If a relationship is not found for the updated table name, an error occurs when the DEFERRED_DELETE is performed.

Only one DEFCID parameter may be specified for a DEFERRED_DELETE Process.

COMMIT_COUNT

To override the commit count for deletes that was specified when the job was created, specify:

```
COMMIT_COUNT      value
```

The value can range from zero to the site limit.

COMMIT_MINUTES

To change commit processing from number of deletes to elapsed time, specify:

```
COMMIT_MINUTES      value
```

The value is specified in minutes and will override the commit count. The value can range from 1 to 1440. The process report will reflect the change from the number of updates to elapsed time.

SEARCH

Use a SEARCH statement to search for specific data within an Archive File and place the search results in a data set that you can review online.

The SEARCH statement is typically generated during an online search when you specify that the search is executed in batch, rather than online. However, you can also use the following syntax to submit a batch search request.

```
SEARCH
  NAME  filename
  OUTDSN dsname
  TABLE ( [ cid. ] tblname1, coloperator, colname1, critoperator, criteria
        [, colname2, critoperator, criteria, ... ] )
  [ TABLE ( [ cid. ] tblname2, coloperator, colname1,critoperator, criteria
        [, colname2, critoperator, criteria, ... ] ) ]
  [ TABCON { AND | OR } ]
  [ RESOURCES { I | C | A } ]
  [ CASE { YES | NO } ]
  [ PRINT { YES | NO } ]
     [ DDNAME ddname ]
     [ WRAP_LINE { YES | NO } ]
     [ VAR_CHAR_DELIMITER c ]
     [ NULL_CHARACTER c ]
```

The keywords and operands in the SEARCH statement are:

Archive File Keyword

NAME

The name of the Archive File to search. NAME is required.

filename

The fully qualified name of a cataloged Archive File data set as an explicit value or a pattern, using DB2 LIKE syntax.

Note: To process multiple Archive Files, you must either use a pattern or multiple SEARCH statements (one for each Archive File).

Output Keyword

OUTDSN

The name of an existing sequential data set to which the search results will be written. The data set must have these attributes: RECFM=FB, LRECL=80, and the BLKSIZE value must be a multiple of 80. This keyword and the *dsname* operand must be specified in the SEARCH statement.

dsname

The fully qualified name of the data set.

Criteria Keywords

TABLE

Criteria used to search a table in the Archive File. At least one TABLE keyword and corresponding operands must be included in a SEARCH statement.

cid.tblname
The table name. Prefix with the Creator ID, if needed to identify the table.

coloperator
Specify either AND or OR, regardless of the number of *colname* operands.

AND Search for data that match criteria for all columns.

OR Search for data that match criteria for at least one column.

colname
The name of the column. At least one *colname* operand, with *coloperator*, *critoperator*, and *criteria*, must be used.

critoperator
Operator for criteria. (See following explanation for *criteria*.)

criteria Up to 250 characters of *criteria* per column. If the column has a character, graphic, or binary data type (e.g., CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP or TIMESTAMP WITH TIME ZONE), the criteria must be delimited with single quotes. If the column has a numeric data type (e.g., INTEGER, SMALLINT, BIGINT or DECIMAL), the criteria must not be in quotes.

Separate *criteria* and *critoperator* with a comma.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

- EQ, *criteria*
- NE, *criteria*
- GT, *criteria*
- LT, *criteria*
- GE, *criteria*
- LE, *criteria*
- IN, (*a,b,c,d...*)
- NOT IN, (*a,b,c,d...*)
- IS NULL
- IS NOT NULL
- LIKE, *pattern*
- NOT LIKE, *pattern*
- BETWEEN, *x* AND *y*
- NOT BETWEEN, *x* AND *y*

Processing Keywords

TABCON
Logical operator for combining TABLE criteria specified in the SEARCH statement. If omitted, data must meet criteria for all tables.

AND Search for data that matches criteria for all tables.

OR Search for data that matches criteria for one table.

RESOURCES
Resources used to resolve the search. If omitted, all resources are searched.

I Search indexes only. If the search cannot be resolved using indexes, do not search Archive Files. (Use this method to locate a file on tape without mounting tape files.)

C Search disk and migrated indexes and, if the search cannot be resolved using indexes, search disk and migrated Archive Files, also.

A Search all indexes and, if necessary, all Archive Files, regardless of storage media. This is the default.

CASE Case requirement for search. If omitted, the search is not case-sensitive.

YES Search for data that matches case of criteria.

NO Search for data that matches criteria without regard to case.

Print Keywords

PRINT

Indicate whether to print.

YES Print archived rows that match the criteria.

NO Do not print (default).

DDNAME

DDNAME for the Print data set. (The default DDNAME for a Print file, used if you do not provide a DDNAME keyword, is PSDFRPRT.)

ddname

Specify the DDNAME for the Print file.

WRAP_LINE

The action taken if archived rows are longer than the output lines.

YES Wrap lines.

NO Change output file characteristics to accommodate the longest record in the file (default). This optional setting is appropriate when output is directed to a data set.

VAR_CHAR_DELIMITER

The character used to delimit variable character data:

c ?

!

&

%

#

@

NULL_CHARACTER

The character used to indicate a field is NULL:

c ?

!

&

%

#

@

Example

To search the Archive File, PST.ARC.FIRST, for customers from New Jersey that have year-to-date sales greater than 3000, specify the following:

```
SEARCH  NAME  PST.ARC.FIRST
   OUTDSN  PST.ARCHIVE.SEARCHOUT
   TABLE (PSTARC.CUSTOMERS, AND,
      YTD_SALES, GT, 3000,
      STATE, EQ, 'NJ')
```

RESTORE

Use a RESTORE statement to restore data from one or more Archive Files using one or more Table Maps and optional criteria.

You can specify Archive Files and Table Maps using a list of names or DB2 LIKE syntax. The RESTORE statement can be a convenient way to automate the restoration of archived data. For example, an application might generate a RESTORE statement to restore archived data when conditions require.

You can also use a RESTORE statement without restoring the data. Specify wildcards and the PREVIEW YES parameter to generate a list of matching Archive Files and to indicate which Archive File would have been restored.

```
RESTORE
  ARCHIVE_FILE ( { USEDD | filename1 [, filename2 ] [ ,... ] } )
     [ GROUP  group ]
     [ DESC desc ]
     [ FROMDATE yyyy-mm-dd ]
     [ TODATE yyyy-mm-dd ]
     [ OLDER (n parameter [, n parameter,... ] ) ]
     [ SORT ( [ NAME, { A | D } ] [ GROUP, { A | D } ] [ DESC, { A | D } ]
        [ DATE, { A | D } ] [ ARCHIVEDBY, { A | D } ] ) ]
     [ RESTORE_MULTIPLE { FIRST | ALL } ]
  { TABLE_MAP_DEFINE (parameters) ; | TABLE_MAP ( mapid.name1 [, mapid.name2, ... ] )
     [ TMDEFCID defcreatorid ]
     [ UNKNOWN { ALLOW | FAIL } ]
     [ TM_SORT [ DATE, ] { A | D } ] }
  { CONTROL_PREFIX  value | CONTROL_NAME explicitfilename }
     [ USE_EXISTING_CONTROL { YES | NO } ]
     [ CONTROL_DELETE { YES | NO } ]
  { SUBSET_PREFIX  { value | * } | SUBSET_NAME explicitfilename }
     [ USE_EXISTING_SUBSET { YES | NO } ]
     [ SUBSET_DELETE  { YES | NO } ]
  [ ROWLIST {dsname | (USEDD) } ]
  [ SELECT ( [ cid. ] tblname1, coloperator, colname1, critoperator, criteria
        [, colname2, critoperator, criteria, ... ] ) ]
  [ SELECT ( [ cid. ] tblname2, coloperator, colname1,critoperator, criteria
        [, colname2, critoperator, criteria, ... ] ) ] ...
  [ SQL ( [ cid. ] tblname1, { AND | OR }, sqlclause1 ) ]
  [ SQL ( [ cid. ] tblname2, { AND | OR }, sqlclause2 ) ]...
  [ CASE { YES | NO } ]
  [ START_TABLE [ cid. ] tblname ]
  [ REF_TABLE ( [ cid. ] tblname1, { YES | NO } ) ]
  [ REF_TABLE ( [ cid. ] tblname2, { YES | NO } ) ]...
  [ REL_OVRD ( TABLE [ cid. ] tblname REL relname
     [ SELECT { YES | NO } | Q1 { YES | NO } | Q2 { YES | NO } ] ) ]
  [ RESTORE_MODE  { INS | UPD | BOTH } ]
  [ MAX_RESTORE_ROWS n ]
  [ LOCK_TABLES  { YES | NO } ]
  [ COMMIT_COUNT  n ]
  [ COMMIT_MINUTES  n ]
  [ DISCARD_COUNT n ]
  [ RESTORE_ERROR { STOP | CONTINUE } ]
  [ TM_LIST { YES | NO } ]
[ PREVIEW { YES | NO } ]
[ REPORT_LEVEL { DETAIL | SUMMARY } ]
[ NOT_FOUND_RC4 { YES | NO } ]
[ EMPTY_TABLE_FOUND_RC4 { YES | NO } ]
[ RESTART { YES | NO } ]
```

Archive File parameters

Use the following parameters to specify one or more Archive Files.

ARCHIVE_FILE

The name of one or more Archive Files to be restored. Multiple values must be enclosed in parentheses and separated by commas. (Note that if you specify a single value, parentheses are optional.) NAME is required.

USEDD

Use the file name specified in the PSDFEXTR control card statement. Only one USEDD operand is supported.

filename

The fully qualified name of a cataloged Archive File data set as an explicit value or a pattern, using DB2 LIKE syntax.

You can list Archive File names and patterns in any order. Use the SORT parameter to specify the order in which the Archive Files are processed; by default, the files are processed in ascending order.

If you specify more than one name, or if any name uses a pattern, you must also specify the CONTROL_PREFIX parameter.

GROUP

Group designation for the Archive Files to be restored.

group Group as an explicit value or a pattern, using DB2 LIKE syntax.

DESC Case-sensitive descriptive information for the Archive Files to be restored.

desc Description as an explicit value or a pattern, using DB2 LIKE syntax. Delimit values that include blanks in single (‘ ') or double quotes (“ ”).

FROMDATE

The oldest date in a range of Archive File creation dates. Data from the selected Archive Files is restored. You can use FROMDATE or OLDER, but not both.

yyyy-mm-dd

Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use FROMDATE alone to select all Archive Files created on or after that date.
- Use FROMDATE with TODATE to designate a range.

TODATE

The most recent date in a range of Archive File creation dates. Data from the selected Archive Files is restored. You can use TODATE or OLDER, but not both.

yyyy-mm-dd

Date in ISO, European, or USA format. Archive converts the date to your DB2 default format.

- Use TODATE alone to select all Archive Files created on or before that date.
- Use TODATE with FROMDATE to designate a range

OLDER

An interval used to exclude Archive Files; that is, only data from Archive Files that are older than the specified interval is restored. You can use OLDER, or FROMDATE and TODATE, but not both.

n The number of days, weeks, months, or years in the interval.

parameter

DAY(S)
WEEK(S)
MONTH(S)
YEAR(S)
You can use one or more parameters (DAYS, WEEK, etc.), each with an *n* operand, in any order.

Examples:

- Specify OLDER (1 YEAR) on March 23, 1999, to restore data from Archive Files created on or before March 23, 1998.
- Specify OLDER (2 YEARS, 6 MONTHS) on June 30, 1999, to restore data from Archive Files created on or before December 31, 1996.

SORT The sequence in which Archive Files are matched with file criteria. If you do not specify a SORT parameter, the files are matched in ascending order, by name.

NAME
In order by name.

GROUP
In order by group.

DESC In order by description.

DATE In order by date created.

ARCHIVEDBY
In order by TSO ID of the user that created the Archive File.

For all of the SORT parameters, valid values are:

A In ascending order (default).

D In descending order.

RESTORE_MULTIPLE
Selection of Archive Files for restoration, if more than one matches file criteria. If you do not specify a RESTORE_MULTIPLE parameter, data from all matching files is restored.

FIRST Restore data from the first file that matches file criteria.

ALL Restore data from all files that match file criteria.

Table Map parameters

Use the following parameters to specify one or more Table Maps.

TABLE_MAP
The name of one or more Table Maps used to restore data in one or more Archive Files. Multiple values must be enclosed in parentheses and separated by commas. (Note that if you specify a single value, parentheses are optional.) TABLE_MAP is required.

mapid.name
The fully qualified name of a Table Map as an explicit value or a pattern, using DB2 LIKE syntax. If you specify more than one Table Map name, or if any name uses a pattern, data in an Archive File is restored using the first Table Map with the same number of tables and the same table names as the archived data. Columns for each table in the Table Map must be compatible with those in the Archive File.

Use the TM_SORT parameter to specify the order in which Table Maps are matched with Archive Files; by default, they are matched in ascending order.

Use the TM_LIST parameter to list the Table Maps used in the process, and those not used.

TABLE_MAP_DEFINE

The Optim online process generates this parameter when it creates a RESTORE job for batch execution.

When you create a RESTORE job outside of the Optim online process, the best practice is to use the TABLE_MAP parameter to refer to a named Table Map in the Optim Directory.

Place the TABLE_MAP_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See TABLE_MAP_DEFINE Parameters for the allowable parameters.)

TMDEFCID

Override for the destination Creator ID for the selected Table Map.

defcreatorid

The default Creator ID

UNKNOWN

Processing for tables referenced in the Table Map that are unknown.

ALLOW

Bypass unknown table and continue RESTORE.

FAIL Terminate the RESTORE (default).

TM_SORT

The order in which Table Maps are matched with Archive Files. Omit this parameter to match Table Maps in ascending order, by name.

DATE Match Table Maps by last modified date sequence.

A Match Table Maps in ascending order (default).

D Match Table Maps in descending order.

Control File parameters

Use the following parameters to specify the Control File. All parameters required to create the control file are determined automatically by the batch utility or supplied as installation defaults.

CONTROL_PREFIX

The prefix used to generate a data set name for the Control File. A data set qualifier in the form ".Cnnnnn" is appended to create a unique Control File name for each restore operation. This parameter is required if you specify multiple Archive File names or a pattern for the NAME parameter, or if the CONTROL_NAME parameter is not specified.

value A maximum of 32 characters.

CONTROL_NAME

The name of the Control File for each restore operation. This parameter is required unless CONTROL_PREFIX is specified, but is allowed only when the RESTORE_ERROR_STOP has been specified or assumed by default.

explicitfilename

The fully qualified name of the Control File used for each restore operation. An existing data set is overwritten.

USE_EXISTING_CONTROL

The action taken when the Control File already exists. Specify:

YES Use the existing Control File.

NO Delete the existing Control File and reallocate it. This is the default.

CONTROL_DELETE

Disposition of a Control File if no errors occur. This parameter applies only if CONTROL_PREFIX is also present. Omit the CONTROL_DELETE parameter to retain the Control File.

YES Delete the Control File after the Restore Process is completed.

NO Retain the Control File after the Restore Process is completed

Subset File parameters

Use the following parameters to specify the Subset File. All parameters required to create the subset file are determined automatically by the batch utility or supplied as installation defaults.

SUBSET_PREFIX

A prefix used to generate a data set name for a file containing the subset of data described by selection criteria. A data set qualifier in the form ".Snnnnn" is appended to create a unique subset file name. This parameter is required if a pattern is used for the Archive File name or SUBSET_NAME is not specified.

value A maximum of 32 characters.

* The value specified for CONTROL_PREFIX is used. Different data set qualifiers prevent a DSN conflict.

SUBSET_NAME

The name of the file containing the subset of data described by selection criteria. This parameter is required unless SUBSET_PREFIX is specified. RESTORE_ERROR_STOP must be specified or assumed by default.

explicitfilename

Fully qualified name of subset file. An existing data set is overwritten; a new data set is allocated and used.

USE_EXISTING_SUBSET

The action taken when the subset file already exists.

YES Use the existing subset file.

NO Delete the existing subset file and reallocate it. This is the default.

SUBSET_DELETE

Disposition of the subset file if no errors occur during the Restore Process. This parameter applies only if SUBSET_PREFIX is also present. Omit the SUBSET_DELETE parameter to retain the subset file.

YES Delete the subset file after the process is completed.

NO Do not delete the subset file (default).

Selection Criteria parameters

Use the following parameters to select the data to restore.

ROWLIST

The name of a Point-and-Shoot file used to select data for restoration.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

dsname

The fully qualified name of a sequential file or partitioned data set with member name.

Note: Specify $$TEMP to indicate a PSDFPNS DD * data set.

(USEDD)

Use the DDNAME PSDFPNS to specify the Point-and-Shoot file. Only one USEDD operand is supported.

SELECT

Criteria used to select data. If you omit the SELECT parameter, all rows in the Archive File are restored.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

cid.tblname

The table name. Prefix with Creator ID, if needed to identify the table.

coloperator

Operator used to combine criteria for the table. One of the following must be indicated, regardless of the number of *colname* operands.

AND Select data that matches criteria for all columns.

OR Select data that matches criteria for at least one column.

colname

The name of the column. At least one *colname* operand, with *coloperator*, *critoperator*, and *criteria*, must be used.

critoperator

Operator for criteria. (See following explanation for *criteria*.)

criteria Up to 250 characters of *criteria* per column. If the column has a character, graphic, or binary data type (e.g., CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE), the criteria must be delimited with single quotes. If the column has a numeric data type (e.g., INTEGER, SMALLINT, BIGINT or DECIMAL), the criteria must not be in quotes. If the Archive File has dense indexes that you want to use, the criteria must be case-sensitive. See the CASE parameter for more information. Separate *criteria* and *critoperator* with a comma.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

- EQ, *criteria*
- NE, *criteria*
- GT, *criteria*
- LT, *criteria*
- GE, *criteria*
- LE, *criteria*
- IN, (*a,b,c,d...*)
- NOT IN, (*a,b,c,d...*)
- IS NULL
- IS NOT NULL
- LIKE, *pattern*
- NOT LIKE, *pattern*
- BETWEEN, *x* AND *y*
- NOT BETWEEN, *x* AND *y*

SQL SQL WHERE criteria used to select data from a table in the Archive File.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

AND Select data that matches selection criteria and SQL WHERE clause for table (default).

OR Select data that matches selection criteria or SQL WHERE clause for table.

sqlclause
Specify criteria as an SQL WHERE clause. Do not use quotes or parentheses; clause ends with statement (at close parenthesis).

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

CASE Case requirement for matching selection criteria. Omit the CASE parameter to select data without regard to case. The value for this parameter must be YES if the Archive File has dense indexes that you want to use. Specifying the value NO or omitting this parameter causes any dense indexes to be ignored.

YES Select data that match case of criteria.

NO Select data that match criteria without regard to case. This is the default.

START_TABLE
Override for the Start Table in Access Definitions used to create the Archive Files.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

REF_TABLE
If a selective restore is used, you can identify tables to use as reference tables in the Restore Process. All rows in a reference table are restored, unless selection criteria are specified for the table. By default, the reference table specifications in the Access Definition used to create the Archive File are used in a selective restore.

You can add reference table specifications or remove an existing reference table specification for tables. Use the REF_TABLE parameter for each table.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

YES Table is a reference table.

NO Table is not a reference table.

REL_OVRD
Overrides the “select” flag and Q1/Q2 values for a relationship, as defined in the Access Definition. The REL_OVRD parameters must be enclosed in parentheses and separated by spaces.

See Select Relationships in the *Common Elements Manual* for a detailed description of how Optim uses the following relationship parameters.

TABLE
Name of the child table in the relationship (required).

Note: Prefix with the Creator ID, if needed to identify the table. If you omit the Creator ID, the default Creator ID from the Access Definition is used.

REL The relationship name (required).

SELECT
Indicates use of the relationship in processing (optional).

YES Use the relationship.

NO Ignore the relationship.

Q1 The Q1 value for the relationship (optional).

Q2 The Q2 value for the relationship (optional).

Note: The SELECT, Q1, and Q2 parameters are optional. If omitted, the corresponding Access Definition values apply.

Processing parameters
Use the following parameters to specify processing options for the Restore Process.

RESTORE_MODE
The method used in the Restore Process. This parameter must be used if a Site Option allows users to select the mode.

INS Data rows are inserted.

UPD Data rows are updated.

BOTH Data rows are both inserted and updated. If a restored row matches a destination row, the destination row is updated; if the restored row does not match a destination row, it is inserted.

MAX_RESTORE_ROWS
The maximum number of rows to be restored. The Restore Process is terminated if the number of restored rows exceeds this limit.

n A number from 1 to 4,294,967,295.

LOCK_TABLES
Indicate whether to lock tables during processing. This parameter is not allowed if the site option prevents users from locking tables.

YES Lock tables.

NO Do not lock tables.

If you do not use the LOCK_TABLES parameter, tables are not locked during processing.

COMMIT_COUNT
The number of rows processed between commit operations. Omit the COMMIT_COUNT parameter to use the site limit.

n A number from 1 to the site limit.

COMMIT_MINUTES
The number of minutes of elapsed time between commit operations. A COMMIT_MINUTES specification overrides COMMIT_COUNT.

n A number from 1 to 1440.

DISCARD_COUNT
The maximum number of discarded rows per restore. Omit the DISCARD_COUNT parameter to allow an unlimited number of rows to be discarded.

n A number from 1 to 4,294,967,295.

RESTORE_ERROR
Processing when error condition occurs during restoration. This parameter applies to operations and Archive Files subsequently processed while executing the RESTORE statement. It does not affect the execution of subsequent Batch Utility control statements.

STOP Further restore operations for the RESTORE statement are suppressed.

CONTINUE
Further restore operations for the RESTORE statement are executed.

If you do not use the RESTORE_ERROR parameter, processing of the RESTORE statement stops when an error condition occurs.

Reporting

TM_LIST
Reporting for Archive Files and Table Maps referenced in the process. If you omit the TM_LIST parameter, the report is not generated.

YES For each Archive File, generate a list of Table Maps referenced in the process, indicating whether a Table Map is selected or acceptable for use or the reason it is not. The report is written to PSDFASUM.

Note: Use PREVIEW YES with TM_LIST YES to generate the list without restoring the data.

NO Do not generate a list of Archive Files and Table Maps referenced in the process.

PREVIEW
Processing indicator.

YES Execute the RESTORE statement without processing the data. PREVIEW YES generates a list of Archive Files that match wildcard entries in the RESTORE statement and indicates which Archive File(s) would have been restored had the data been processed.

NO Execute the RESTORE and process all data. If you omit the PREVIEW parameter, all processing takes place.

REPORT_LEVEL
The level of detail provided in the Process Report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

NOT_FOUND_RC4
Indicate the action taken and return code (RC) setting when no rows are restored in a selective restore.

Note: The Process Report Summary will also include messages when no rows are restored. This parameter affects only the return code.

YES A warning message will be issued with RC=4 (default).

NO No messages will be issued with RC=0.

EMPTY_TABLE_FOUND_RC4
Indicate the action taken and return code (RC) setting when empty tables have been detected during Restore batch processing.

YES Empty tables will be processed and a warning message will be issued with RC=4.

NO Empty tables will cause termination with an error and RC=12 (default).

RESTART
Indicate whether this is a restart or retry attempt for a RESTORE process that failed earlier. Optim automatically determines whether to perform a restart or a retry.

YES Restart or retry RESTORE process.

NO Do not attempt to restart or retry a RESTORE process (default).

Example

Specify the following to restore rows from the Archive File PSTARC.AF1, using Table Map PSTARC.TM1, and meeting the selection criteria for salesmen in the western territory who are over 21 years old. The following also specifies to use the named Control File and subset file, and to insert data rows.

```
RESTORE
   ARCHIVE_FILE PSTARC.AF1
   TABLE_MAP PSTARC.TM1
   CONTROL_NAME PSTARC.BTCHCNTL.FILE1
   SUBSET_NAME PSTARC.BTCHSUBS.FILE1
      SELECT (PSTARC.SALES, AND, AGE, GT, 21, TERRITORY, EQ, 'West')
      REL_OVRD (TABLE PSTMJG.CUSTOMERS REL CUST Q2 YES)
   RESTORE_MODE INS
```

Note: To execute a RESTORE statement without restoring the data, use wildcards and the PREVIEW YES parameter to generate a list of matching Archive Files and to indicate the Archive File that would otherwise be restored.

RESTORE Override Keywords

When you submit the job, you can use overrides to apply different Archive Files to a single set of tables or a single Archive File to multiple sets of tables using common Restore Process JCL. You can override the default destination Creator ID defined in the Table Map used for the Restore Process.

Any Archive File can be used in a Restore Process as long as at least one table name in the file matches one table name in the Table Map. The Creator IDs do not have to match. If any table does not match in the Table Map, it is not included in the process. Use the PSDFOVRD DD statement to provide the desired overrides. These overrides are intended to be used only in a batch Restore Process. Do not use them in any other process.

ARCHIVE_DSN

To specify a new data set name for the Archive File, specify:

```
ARCHIVE_DSN     data.set.name
```

This override allows you to use one set of saved JCL to restore Archive Files with different names.

COMMIT_COUNT

To override the commit count that was specified when the job was created, specify:

```
COMMIT_COUNT      value
```

The value can range from zero to the site limit.

COMMIT_MINUTES

To change commit processing from number of updates to elapsed time, specify:

```
COMMIT_MINUTES      value
```

The value is specified in minutes and will override the commit count. The value can range from 1 to 1440. The process report will reflect the change from the number of updates to elapsed time.

DEFCID

To override the default destination Creator ID specified on the Table Map, specify:

```
DEFCID cid
```

Where *cid* is the default Creator ID. This Creator ID applies to destination tables that are not explicitly qualified in the Table Map. Only one DEFCID parameter may be specified for a Restore Process.

RESTORE_SEL

To add new selection criteria, change, or delete existing selection criteria, specify:

```
RESTORE_SEL [ cid.] table   column  [ selcriteria ]
```

cid.table
The table name must be specified. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

column The column name must be specified.

selcriteria
The selection criteria for the column must conform to the format required in the Specify Selection Criteria for AD panel (see the *Common Elements Manual* for further information). The maximum length is 53 characters. Omit this operand to delete existing selection criteria for a column.

RESTORE_SQL

To add an SQL clause, change, or delete an existing SQL clause, specify:

```
 RESTORE_SQL  [ cid.] table  AND | OR  [ where ]
```

cid.table
The table name must be specified. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

AND Select data that matches selection criteria and SQL WHERE clause for table. Default.

OR Select data that matches selection criteria or SQL WHERE clause for table.

where The WHERE clause must conform to the requirements specified for the Specify SQL WHERE Clause panel (see the *Common Elements Manual* for further information). The keyword WHERE is not required as part of the clause. To delete an existing SQL clause for a table, omit this operand.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

VAR

To override the default value of a substitution variable assigned in the Access Definition, specify:

```
VAR   varname   value
```

varname
The name of the substitution variable assigned in the Access Definition. A colon (:) in front of *VarName* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

RESTORE_START_TABLE

To change the name of the start table for a selective restore, specify:

```
RESTORE_START_TABLE  [ cid.] table
```

cid.table
: The table name overrides the existing start table. If the Creator ID (*cid*) is omitted, the Creator ID of the existing start table will be used.

SUBSET_DSN

To override the data set name for the subset Archive File, specify:

```
SUBSET_DSN     data.set.name
```

If adding selection criteria or an SQL clause to change a full restore to a selective restore, this override must be present or an error occurs.

UNKNOWN

If any tables referenced in the Table Map do not exist at the time of the Restore Process, specify:

```
UNKNOWN    { FAIL | ALLOW }
```

FAIL Terminate the Restore Process if any tables or relationships in the Archive File are unknown. A message in the report identifies the first unknown object. FAIL is the default.

ALLOW
Bypass any unknown tables and relationships and continue with the Restore Process. A message in the report identifies the unknown object.

UNKNOWN ALLOW is used with the DEFCID override. Changing the default Creator ID may result in destination tables that do not exist. Use UNKNOWN ALLOW to skip these “unknown” tables.

Store Overrides

You can store batch parameters in a sequential file or a partitioned data set, rather than enter job control directly into the jobstream. However, the parameters must be the only data in the file.

SUBSET

Use the SUBSET statement to create a subset file from an existing archive file. The original archive file remains intact in the directory.

The subset file is an archive file that contains a subset of related rows from the source archive file and the objects needed to recreate the database tables. You can use the subset file to restore previously archived rows in a database.

There is a growing need to ensure information privacy and preserve data for audit or legal purposes. It is equally important to establish policies to dispose of information at the end of its lifecycle. Features of the Subset process enable you to meet these requirements. You can use Subset to select rows from the original archive and create a file with a retention period that is different from the source archive file. When you create subset file you can specify a Point-and-Shoot list, override variables used in the source archive file, change relationship properties and control whether a table is processed as a reference table. Additionally you can register the subset file in the archive directory, create an index for it, and add it to a collection.

If security is active at your site, you can assign security status for any files created using the SUBSET process and cataloged in the archive directory.

```
SUBSET
  ARCHIVE_NAME {explicitfilename | (USEDD)}
  SUBSET_FILE(File Allocation Parameters)
    [ COPY_ARCHIVE_INFO { YES |NO }]
  [ REF_TABLE ([ cid. ] tblname1 { YES | NO })]
  [ ROWLIST {dsname| (USEDD) } ]
  [ POINT_SHOOT_ERROR { STOP | CONTINUE | SUBSET }
  [ SELECT ( [ cid. ] tblname1,coloperator,colname1,   critoperator,criteria
             [, colname2,critoperator,criteria,... ] ) ]
  [ SELECT ( [ cid. ] tblname2,coloperator,colname1, critoperator,criteria
             [, colname2,critoperator,criteria,... ] ) ]...
  [ SQL ( [ cid. ] tblname1, { AND | OR }, sqlclause1 ) ]
  [ SQL ( [ cid. ] tblname2, { AND | OR }, sqlclause2 ) ]...
  [ CASE { YES | NO } ]
  [ INDEX_FILE (File Allocation Parameters)]
  [ DROP_INDEX_COLUMN( [ cid. ] table, column1 [,  column2, ... ] ) ]
  [ DROP_INDEX_TABLE( [ cid. ] table1 [, [ cid. ] table2,... ] ) ]
  [ DROP_INDEX_ALL{ YES | NO } ]
  [ SPARSE_INDEX( [ cid. ] table, column1 [, column2, ...] ) ]
  [ DENSE_INDEX( [ cid. ] table, column1 [, column2, ... ] ) ]
  [ VAR (varname, value)]
  [ SECURITY_STATUS { PUBLIC | READONLY | PRIVATE } ]
  [ GROUP group ]
  [ DESC desc ]
  [ RETENTION_PERIOD{ NOLIMIT | nD | nY | yyyy-mm-dd | yyyy.ddd | PERM | NEVER } ]
  [ SKIP_ARC_CATALOG  { YES | NO }]
  [ REPLACE_ARC_DIR { YES | NO }]
  [ COLLECTION (cid.collectionname [,cid.collectionname2,...])]
  [ START_TABLE [ cid. ] tblname ]
  [ REL_OVRD (TABLE [ cid.] tblname, REL relname
                 [ SELECT { YES | NO } | Q1 { YES | NO } | Q2 { YES | NO } ] ) ]]
  [ REPORT_LEVEL { DETAIL | SUMMARY } ]
  [ NOT_FOUND_RC4 { YES | NO } ]
```

ARCHIVE_NAME

The name of the source archive file from which the subset file will be created. ARCHIVE_NAME is required.

explicitfilename

The fully qualified name of a cataloged archive file.

(USEDD)

Use the DDNAME PSDFEXTR. Only one USEDD operand is supported.

SUBSET_FILE

Name of the output file that will contain the subset of rows from the source archive file. This parameter is required. See File Allocation Parameters.

COPY_ARCHIVE_INFO

Use this parameter to create the subset file with the same attributes as the source archive file.

YES Copy the source archive file attributes EXPIRATION_DATE, SECURITY_STATUS, GROUP, DESC and RETENTION_PERIOD and use these attributes for the subset file created by this process.

Note: EXPIRATION_DATE cannot be overridden. All of the remaining attributes can be overridden using the respective parameters.

NO Do not copy the source archive file attributes, except for EXPIRATION_DATE which is always copied and used for the subset file. This is the default.

Processing parameters

Use the following parameters to provide processing options for the archive file.

REF_TABLE

Use this parameter to identify tables to use as reference tables in the subset process. All rows in a reference table are selected, unless selection criteria are specified for the table. By default, the reference table specifications in the Access Definition used to create the source Archive File are used. You can add reference table specifications or remove an existing reference table specification for tables. Use the REF_TABLE parameter for each table.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

YES Table is a reference table.

NO Table is not a reference table.

ROWLIST
The name of a Point-and-Shoot file used to select data for subsetting.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

dsname
The fully qualified name of a sequential file or partitioned data set with member name.

(USEDD)
Use the DDNAME PSDFPNS to specify the Point-and-Shoot file. Only one USEDD operand is supported.

POINT_SHOOT_ERROR
Indicate action to be taken when processing the Point-and-Shoot file results in errors.

STOP Stop the run if the Point-and-Shoot file contains invalid keys or is empty. This is the default.

CONTINUE
Continue the run if invalid keys are found, but do not use the Point-and-Shoot file.

SUBSET
If invalid keys are found, continue the run with the valid keys only. If errors other than invalid keys are found, stop the run.

SELECT
Criteria used to select data. If you omit the SELECT parameter, all rows in the archive file are copied to the subset file.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

cid.tblname
The table name. Prefix with Creator ID, if needed to identify the table.

coloperator
Operator used to combine criteria for the table. One of the following must be indicated, regardless of the number of *colname* operands.

AND Select data that matches criteria for all columns.

OR Select data that matches criteria for at least one column.

colname
The name of the column. At least one *colname* operand, with *coloperator*, *critoperator*, and *criteria*, must be used.

critoperator
Operator for criteria. (See following explanation for *criteria*.)

criteria Up to 250 characters of *criteria* per column.

If the column has a character, graphic, or binary data type (e.g., CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE), the criteria must be delimited with single quotes.

If the column has a numeric data type (e.g., INTEGER, SMALLINT, BIGINT or DECIMAL), the criteria must not be in quotes.

If the archive file has dense indexes that you want to use, the criteria must be case-sensitive. See the CASE parameter for more information.

Separate *criteria* and *critoperator* values with a comma.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

- EQ, *criteria*
- NE, *criteria*
- GT, *criteria*
- LT, *criteria*
- GE, *criteria*
- LE, *criteria*
- IN, (*a,b,c,d...*)
- NOT IN, (*a,b,c,d...*)
- IS NULL
- IS NOT NULL
- LIKE, *pattern*
- NOT LIKE, *pattern*
- BETWEEN, *x* AND *y*
- NOT BETWEEN, *x* AND *y*

SQL SQL WHERE criteria used to select data from a table in the source archive file.

Note: If you specify ROWLIST and SELECT or SQL criteria, only rows that meet the SELECT or SQL criteria are matched against the ROWLIST.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

AND Select data that matches selection criteria and SQL WHERE clause for table (default).

OR Select data that matches selection criteria or SQL WHERE clause for table.

sqlclause
Specify criteria as an SQL WHERE clause. Do not use quotes or parentheses; clause ends with statement (at close parenthesis).

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

CASE Case requirement for matching selection criteria. Omit the CASE parameter to select data without regard to case. The value for this parameter must be YES if the archive file has dense indexes that you want to use. Specifying the value NO or omitting this parameter causes any dense indexes to be ignored.

YES Select data that match case of criteria.

NO Select data that match criteria without regard to case. This is the default.

INDEX_FILE
Create an Index file for dense indexes. If omitted, and a dense index is to be created, an error occurs. Use the parameters shown in Chapter 5, "File Allocation Parameters," on page 139

DROP_INDEX_COLUMN
Drop the index for one or more columns in a table in the Access Definition. (If the Access Definition does not have an index for the specified column, an error occurs.) You can specify multiple DROP_INDEX_COLUMN parameters, but duplicate column names cause an error.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns with an index to be dropped.

DROP_INDEX_TABLE
Drop all indexes for the specified tables in the Access Definition. You can specify multiple DROP_INDEX_TABLE parameters, but duplicate table names cause an error.

cid.table
The name of one or more tables with indexes to be dropped. If you omit the Creator ID, the default Creator ID is used.

DROP_INDEX_ALL
Drop all indexes for all tables in the Access Definition.

YES Drop all indexes.

NO Do not drop all indexes (default).

SPARSE_INDEX
Create a sparse index for one or more columns in the specified table. If the Access Definition includes a sparse index for the specified column, an error occurs unless the index has been dropped using a prior DROP_INDEX parameter.

You can use multiple SPARSE_INDEX parameters, but duplicate column names cause an error. You can specify a sparse index for a column that has a dense index.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns to be indexed.

DENSE_INDEX
Create a dense index for one or more columns in the specified table. If the Access Definition defines a dense index for the specified column, an error occurs unless the index has been dropped using a prior DROP_INDEX parameter.

You can use multiple DENSE_INDEX parameters, but duplicate column names cause an error. You can create a dense index for a column that also has a sparse index.

cid.table
The name of the table. If you omit the Creator ID, the default Creator ID is used.

column The name of one or more columns to be indexed.

VAR Override for the default value of a substitution variable assigned in the Access Definition.

If a default value was not defined in the Access Definition, you must use the VAR parameter to specify a value, or an error occurs.

varname
The name of the substitution variable assigned in the Access Definition. A colon (:) before *varname* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, or TIMESTAMP column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

SECURITY_STATUS
Security status to be assigned to the subset file. If archive security is not currently activated, the assigned status becomes effective when security is activated for the site. Omit this parameter to use the site default for archive security (or PUBLIC, if the site default is not specified).

PUBLIC
All can use and modify.

READONLY
All can use, but only the Archive Administrator or the owner can modify.

PRIVATE
Only the Archive Administrator or the owner can use and modify.

GROUP
Group designation for the subset file.

group Up to 8-character group designation, enclosed in single or double quotes if using blanks or special characters.

DESC Description of the subset file.

desc Up to 40-character description, enclosed in single or double quotes.

SKIP_ARC_CATALOG
Indicate whether to create an archive Directory entry for the subset file.

YES Skip creating an entry for the subset file or the duplicate archive file. This is the default.

NO Create an entry for the subset file.

DUP Skip creating an entry for the duplicate archive file.

REPLACE_ARC_DIR
The action taken if an Archive Directory entry already exists for the file created by this process.

Note: If you specify SKIP_ARC_DIR YES, any specification for REPLACE_ARC_DIRECTORY is ignored.

YES Replace the existing entry.

NO Do not replace any existing archive directory entry. This is the default.

RETENTION_PERIOD
The retention period for the subset file. You cannot delete or overwrite the subset file, its associated Archive Directory entry, or any associated archive index file until after the specified retention period.

This parameter is valid only when the Site Option, **Specify Arc File Retention**, is set to User. If omitted, when the Site Option allows user specification, no retention period is assigned.

NOLIMIT
No retention period is assigned.

*n***D** The number of days to retain the subset file.

*n***Y** The number of years to retain the subset file.

*yyyy***-***mm***-***dd***,** *yyyy/mm/dd***, or** *yyyy.ddd*
An explicit date, after which you can delete or overwrite the subset file. Enter a year from 1900-2155. For *ddd* values, enter a day from 000-366.

PERM, NEVER, 1999.365, 1999.366,
1999/12/31, or 99/12/31
The retention period does not expire.

Note: Any archive file assigned a retention period over 9999 days or 27 years is considered permanent, and can only be deleted or overwritten if you reduce the retention period using the ALTER statement of the IBM Utility program, IDCAMS.

SECURITY_STATUS
Security status to be assigned to the subset file. If archive security is not currently activated, the assigned status becomes effective when security is activated for the site. Omit this parameter to use the site default for archive security (or PUBLIC, if the site default is not specified).

PUBLIC
All can use and modify.

READONLY
All can use, but only the Archive Administrator or the owner can modify.

PRIVATE
Only the Archive Administrator or the owner can use and modify.

COLLECTION
If appropriate, specify the name of the Archive Collection(s) to which the subset file should be added. Each collection is identified by a Collection ID of up to eight characters, and a Collection Name of up to 12 characters, such as CID1.COLLECTION. You can specify multiple collections, but duplicate names cause an error.

Note: If you specify the COLLECTION parameter you must also specify SKIP_ARC_CATALOG NO.

cid The Collection ID.

collectionname
The Collection Name.

START_TABLE
Override for the Start Table in the Access Definition used to create the source archive file.

cid.tblname
Table name. Prefix with Creator ID, if needed to identify the table.

REL_OVRD
Overrides the “select” flag and Q1/Q2 values for a relationship, as defined in the Access Definition. The REL_OVRD parameters must be enclosed in parentheses and separated by

spaces. See **Select Relationships** in the *Common Elements Manual* for a detailed description of how Optim uses the following relationship parameters.

TABLE
Name of the child table in the relationship (required).

Note: Prefix with the Creator ID, if needed to identify the table. If you omit the Creator ID, the default Creator ID from the Access Definition is used.

REL The relationship name (required).

SELECT
Indicates use of the relationship in processing (optional).

YES Use the relationship.

NO Ignore the relationship.

Q1 The Q1 value for the relationship (optional).

Q2 The Q2 value for the relationship (optional).

Note: The SELECT, Q1, and Q2 parameters are optional. If omitted, the corresponding Access Definition values apply.

REPORT_LEVEL
The level of detail in the process report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

NOT_FOUND_RC4
Indicate the action taken and return code (RC) setting when no rows meet the selection criteria during subset batch processing.

YES A warning message will be issued and the program will end with RC=4 (default).

NO The program will end with RC=0.

Example

Use the following statement to create a subset file, named DEPT14.OCT2006.CUST.NJ, from the existing archive file, DEPT14.OCT2006.CUST.

```
SUBSET
   ARCHIVE_NAME  DEPT14.OCT2006.CUST
   SUBSET_FILE  (DSNAME  DEPT14.OCT2006.CUST.NJ  MODE REP)
   SELECT  (DEPT14.CUSTOMERS, OR, STATE, EQ, 'NJ')
   CASE YES
```

EXTRACT

Use an EXTRACT statement to create an Extract File that contains the selected set of related rows from one or more tables and, if requested, the object definitions for those tables. The Extract File is used as input to the Move, Insert, Load, Create, and Convert processes. The Extract File can be used repeatedly and simultaneously by many users.

The EXTRACT statement requires an Access Definition defined in the Optim Directory. You can use keywords to override parameters in the Access Definition and to provide parameters similar to those for the online Extract Process.

Note: If Optim is APF-authorized (a requirement to use an unload utility), to extract IMS data using an EXTRACT statement, you must add the Optim program library (SFOPLLIB) to the IMS Program Libraries list in the IMS environment definition. Refer to the *Move User Manual* Definitions section.

```
EXTRACT
 { ACCESS_DEFINITION_DEFINE (parameters) ; |
  ACCESS_DEFINITION group.user.name
   [ DEFAULT_CID cid ]
   [ UNKNOWN { FAIL | ALLOW } ]
   [ SELECT ( [ cid. ]table1,coloperator,column1
     { [ ,critoperator,criteria] | [,DELETE ] }
     [ ,column2{ [ ,critoperator,criteria] | [,DELETE ] } ... ] ) ]
   [ SELECT ( [ cid. ]table2,coloperator,column1
     {critoperator,criteria ] | [,DELETE ] }
     [ ,column2 { [ ,critoperator,criteria  ] | [,DELETE ] } ... ] ) ] ...
   [ SQL ( [ cid. ] table1 [, [ /correlation/ ] whereclause1] ) ]
   [ SQL ( [ cid. ] table2 [, [ /correlation/ ] whereclause2] ) ] ...
   [ VAR ( varname,value ) ]
   [ POINT_SHOOT_DSN explicitfilename ]
   [ POINT_SHOOT_ERROR { STOP | CONTINUE |SUBSET } ]
   [ OBJECT_DEFS { YES | NO } ]
   [ INCLUDE_OBJECT (obj1,obj2,... ) ]
   [ EXCLUDE_OBJECT (obj1,obj2,... ) ]
   [ INCLUDE_DATA { YES | NO } ]
   [ ROW_SELECT_BY { ROWLIST | BOTH } }
 EXTRACT_FILE ( File Allocation Parameters )
 [ UNLOAD_UTILITY { BMC | IBM | CDB | CDBO }
   [ IMAGE_COPY
    ( MODE { L | A | B | D }
    [ DATE yyyy-mm-dd ]
    [ TIME hh.mm.ss ]
    [ DSNAME dsname ] ) ]
   [ PARTITIONS ( part1 [, part2 [ , ... ] ) ][ OTHER_PARTITIONS_SAME { YES | NO }] ]
  [ WITH_UR { YES | NO } ]
  [ ROW_LIMIT n ]
  [ DEFAULT_KEY_LIMIT n ]
  [ SKIP_EXT_CATALOG  { YES | NO } ]
  [ REPLACE_EXT_DIR { YES | NO } ]
  [ CONVERT_PERFORM { YES | NO }
   [ CONVERT_SORT { YES | NO } ]
   [ CONVERT_MAX_DISCARDS n ]
   [ CONVERTED_FILE (File Allocation Parameters) ]
   [ CONTROL_FILE (File Allocation Parameters) ]
   { TABLE_MAP mapid.name | TABLE_MAP_DEFINE (parameters); }
   [ AGING (parameters) ] ]
  [ REPORT_LEVEL { DETAIL | SUMMARY } ]
```

Access Definition Keywords

Use the following keywords to specify the Access Definition and override or augment various parameters in the Access Definition.

ACCESS_DEFINITION

The name of the Access Definition. ACCESS_DEFINITION must be included in the EXTRACT statement and must precede any keywords that override parameters in the Access Definition (e.g., SELECT, SQL, DEFAULT_CID).

group.user.name

The three-part Access Definition name.

ACCESS_DEFINITION_DEFINE

The Optim online process generates this keyword when it creates an EXTRACT job for batch execution.

When you create an EXTRACT job outside of the Optim online process, the best practice is to use the ACCESS_DEFINITION keyword to refer to a named Access Definition in the Optim Directory.

Place the ACCESS_DEFINITION_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See ACCESS_DEFINITION_DEFINE Parameters for the allowable parameters.)

Note: The ADNAME keyword is generated when you specify a named Access Definition in the online process. This keyword is for documentation purposes only.

DEFAULT_CID

Override for the default Creator ID specified in the Access Definition. When used, this keyword must precede any keyword values that require a default Creator ID (e.g., SELECT).

cid The default Creator ID.

UNKNOWN

The action taken if the Access Definition references unknown tables or relationships.

FAIL Terminate processing if any tables or relationships named in the Access Definition are unknown (default).

ALLOW

Bypass unknown tables and relationships and continue processing.

Selection Criteria

SELECT

Selection criteria for rows in a table. Use *critoperator* and *criteria* to augment Access Definition selection criteria (according to *coloperator*) or to override any existing Access Definition selection criteria for the specified column. Use DELETE, instead of *critoperator* and *criteria*, to bypass existing Access Definition selection criteria for the specified column.

cid.table

The table name. If you omit the Creator ID, the default Creator ID is used.

coloperator

The operator used to combine selection criteria for multiple columns in a table. You must specify one of the following:

AND Select data that matches criteria for all columns.

OR Select data that matches criteria for at least one column.

column The name of the column. You must specify at least one *column* operand.

critoperator

Operator for criteria. (See the following explanation for *criteria*.)

criteria Up to 250 characters of *criteria* per column. If the column has a character, graphic, or binary data type (e.g., CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE), the criteria must be delimited with single quotes. If the column has a numeric data type (e.g., INTEGER, SMALLINT, BIGINT or DECIMAL), the criteria must not be in quotes.

Separate *criteria* and *critoperator* with a comma.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

- EQ, *criteria*
- NE, *criteria*
- GT, *criteria*

- LT, *criteria*
- GE, *criteria*
- LE, *criteria*
- IN, (*a,b,c,d...*)
- NOT IN, (*a,b,c,d...*)
- IS NULL
- IS NOT NULL
- LIKE, *pattern*
- NOT LIKE, *pattern*
- BETWEEN, *x* AND *y*
- NOT BETWEEN, *x* AND *y*

DELETE

Bypass existing selection criteria in the Access Definition for the column. If you use DELETE, you must omit *critoperator* and *criteria* for the column.

SQL An SQL WHERE clause for the specified table. Overrides any SQL WHERE clause specified in the Access Definition. (Omit *whereclause* to bypass an Access Definition SQL WHERE clause for the specified table.) You can specify multiple SQL statements, but duplicate table names cause an error.

cid.table

The table name. If you omit the Creator ID, the default Creator ID is used.

/correlation/

The optional correlation name must be enclosed in slashes and must immediately precede *whereclause*. Overrides any correlation name specified in the Access Definition.

whereclause

Lines of criteria as an SQL WHERE clause. Break each line at a blank, start anywhere on the next line, do not use quotes or parentheses, and end the clause with a close parenthesis.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

VAR Override for the default value of a substitution variable assigned in the Access Definition. If a default value was not defined in the Access Definition, you must use the VAR keyword to specify a value, or an error occurs.

varname

The name of the substitution variable assigned in the Access Definition. A colon (:) in front of *VarName* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

POINT_SHOOT_DSN

The name of a Point-and-Shoot file used to select data to be extracted. Overrides any Point-and-Shoot file specified in the Access Definition.

explicitfilename
The fully qualified name of the Point-and-Shoot file.

Note: You can also specify a Point-and-Shoot file by placing a PSDFPNS DD * data set in the job stream, containing key values to be processed. In this case, omit the POINT_SHOOT_DSN keyword.

POINT_SHOOT_ERROR
Indicate action to be taken when processing the Point-and-Shoot file results in errors.

STOP Stop the run.

CONTINUE
Continue the run without using the Point-and-Shoot file.

SUBSET
If invalid keys are found, continue the run with the valid keys only. If errors other than invalid keys are found, stop the run.

Object Definitions

OBJECT_DEFS
Indicate whether to extract object definitions (e.g., primary keys, relationships, indexes, views, synonyms, aliases, procedures, triggers, and user-defined types and functions).

YES Extract object definitions of all types.

NO Do not extract object definitions (default).

INCLUDE_OBJECT
Include specific object types by name if the OBJECT_DEFS keyword is NO. Specify any of the following values in any order within parentheses:

PKREL
Primary keys and relationships

INDEX
Indexes

VIEW Views

ALIAS
Aliases

SYN Synonyms

FPROC
Column Field Procedure Names

TRIG Triggers

UDEF User Defined Types and Functions

SPROC
Stored Procedures

EXCLUDE_OBJECT
Exclude specific object types by name if the OBJECT_DEFS keyword is YES. Specify any of the following values in any order within parentheses:

PKREL
Primary keys and relationships

INDEX
Indexes

VIEW Views

ALIAS
Aliases

SYN Synonyms

FPROC
Column Field Procedure Names

TRIG Triggers

UDEF User Defined Types and Functions

SPROC
Stored Procedures

INCLUDE_DATA
Indicate whether row data from the DB2 tables is to be included in the output file.

YES Include row data (default).

NO Omit row data.

ROW_SELECT_BY
Indicate whether selection criteria or Point-and-Shoot List should be used when Access Definition specifies both for the Start Table.

ROWLIST
Apply Point-and-Shoot List only (default).

BOTH Apply Point-and-Shoot List and selection criteria.

Extract File Keyword

EXTRACT_FILE
EXTRACT_FILE must be included in the EXTRACT statement, and must precede any file processing keywords (e.g., IMAGE_COPY). See File Allocation Parameters

Unload Utility Keywords

Use the following keywords to specify an unload utility:

UNLOAD_UTILITY
The name of the unload utility used to retrieve the data. If omitted, DB2 is used to retrieve the data. This keyword is required to extract data from an image copy.

BMC Use the UNLOAD PLUS utility.

IBM Use the High Performance Unload utility.

CDB Use the Auto-Unload utility.

CDBO
Use the Auto-Online Unload utility.

Note: You must add the appropriate utility DD statements. See the utility documentation for the required DD statements.

IMAGE_COPY
The image copy file to use as input. If omitted, DB2 files are used as input. This keyword is allowed only if the UNLOAD_UTILITY keyword is specified.

IMAGE_COPY keywords must be enclosed in parentheses. MODE is required; other keywords and operands are optional, depending on the value of MODE.

Note: To extract from DB2 image copy data sets in multiple partitions of the same tablespace stored on the same tape volume, you must manually edit the JCL to allocate the data sets. Multiple image copy data sets cataloged on the same tape volume can not be allocated using dynamic allocation. This is a z/OS limitation. If you attempt to use dynamic allocation, the extract process fails with a dynamic allocation error. Refer to Chapter 5, “File Allocation Parameters,” on page 139 for details.

MODE
The image copy file to use:

L Use the latest file.

A Use the first file created on or after the specified DATE and TIME.

B Use the first file created on or before the specified DATE and TIME.

D Use the file named in DSNAME.

DATE Date criteria for the image copy file. Specify the date in this format: ***yyyy-mm-dd***.

TIME Time criteria for the image copy file. Specify the time in this format: ***hh.mm.ss***. If omitted, a value of 00.00.01 is assumed.

DSNAME
The fully qualified name of the image copy file.

PARTITIONS
The numbers of the partitions to be processed. If omitted, all partitions are processed. This keyword applies only if the UNLOAD_UTILITY keyword is specified and the Start Table is partitioned.

part The partition numbers, enclosed in parentheses and separated by commas.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

You may specify partitions in any order. (Partitions that are not specified are not extracted.)

OTHER_PARTITIONS_SAME
Use this operand to indicate whether to process tables in all partitions or only the tables in the partitions specified in the PARTITIONS operand for the Start Table.

YES Limit processing to the tables in the partitions specified in the PARTITIONS operand for the Start Table. This assumes that the rows selected from related tables are physically located in the same partitions as the rows selected from the Start Table.

NO Do not limit processing to the tables in partitions specified in the PARTITIONS operand for the Start Table. If needed, process tables in additional partitions to select rows from related tables. This is the default.

Processing Keywords
Use the following keywords to specify processing options for the Extract File.

WITH_UR
Indicate whether to extract uncommitted data from the database. This keyword is valid only when the Site Option, **Use Uncommitted Reads**, is set to User. Specify:

YES Extract uncommitted data from the database.

Note: If you choose to extract uncommitted data, the relational integrity of the data in the Extract File may be compromised. Use caution if restoring data in any Extract File with uncommitted data.

NO Do not extract uncommitted data from the database. This is the default.

ROW_LIMIT
The maximum number of rows that can be extracted. If omitted, the site limit is used.

n The maximum number of extracted rows. Specify a value in the range 1 to 4,294,967,295

DEFAULT_KEY_LIMIT
The default maximum number of keys used at one time to process a table with key lookup. Applies only if the Access Definition does not specify a limit.

n 1 - 100 (default is 1)

SKIP_EXT_CATALOG
Indicate whether to skip creation of a Directory entry for the extract file.

YES Skip creating a Directory entry for the extract file.

NO Create an entry for the extract file.

REPLACE_EXT_DIR
The action taken if a Directory entry already exists for the specified extract file.

YES Replace the existing entry.

NO Stop the extract process (default).

CONVERT_PERFORM
Indicate whether the Convert Process should be invoked to transform the extracted data after the Extract Process is complete.

YES Convert extracted data.

NO Do not convert extracted data (default).

CONVERT_SORT
Indicate whether to sort rows for destination tables with a cluster index. This keyword applies only if CONVERT_PERFORM is YES. Specify:

YES Sort rows.

NO Do not sort rows (default).

CONVERT_MAX_DISCARDS
The maximum number of discarded rows for the Convert Process. If the maximum value is exceeded, the Convert Process terminates. Omit this keyword to allow an unlimited number of rows to be discarded. This keyword applies only if CONVERT_PERFORM is YES.

n Number in the range 1 - 4,294,967,295.

CONVERTED_FILE
Identify the output file to which the converted data will be written. Use the keywords in File Allocation Parameters to name and allocate the file. If you omit this keyword, the converted data is written back to the Extract File. This keyword applies only if CONVERT_PERFORM is YES

CONTROL_FILE
Identify the Control File. This keyword is required. Use the keywords in File Allocation Parameters to name and allocate the Control File. This keyword applies only if CONVERT_PERFORM is YES.

TABLE_MAP
The name of the Table Map to be used. This keyword applies only if CONVERT_PERFORM is YES.

mapid.name
Fully qualified name of a Table Map in the Optim Directory.

TABLE_MAP_DEFINE
The Optim online process generates this keyword when it creates a CONVERT job for batch execution.

When you create a CONVERT job outside of the Optim online process, the best practice is to use the TABLE_MAP keyword to refer to a named Table Map in the Optim Directory.

Place the TABLE_MAP_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See TABLE_MAP_DEFINE Parameters for the allowable parameters.)

AGING
This keyword indicates that date values in the source columns are to be aged. It provides parameters to be used in the aging process. This keyword applies only if CONVERT_PERFORM is YES.

See AGING Keyword Parameters for detailed information on using this keyword.

REPORT_LEVEL
The level of detail provided in the Process Report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

Example

Use the following statement to create an Extract File named PSTUSER.EXTRACT.CUST, using the Access Definition PSTUSER.AD.CUSTOMERS. This example also uses selection criteria and executes the UNLOAD PLUS utility.

```
EXTRACT
   ACCESS_DEFINITION PSTUSER.AD.CUSTOMERS
   EXTRACT_FILE (DSNAME PSTUSER.EXTRACT.CUST)
   SELECT (PSTUSER.CUSTOMERS, AND, AGE, GR, 21, AREA, EQ, 'WEST')
   UNLOAD_UTILITY BMC
```

EXTRACT Override Keywords

When you submit the job, you can override the default Creator ID, selection criteria, and SQL WHERE clause defined in the Access Definition used for the Extract Process. This is especially useful when several extracts are to be performed for a set of tables that vary only by Creator ID or by some set of selection criteria.

You can override these values using the PSDFOVRD DD statement to point to the desired overrides. (Examples follow the discussion of the overrides.)

UNL_IMAGECOPY_DSN

To override the Image Copy DSN parameter for an unload program, specify:

```
UNL_IMAGECOPY_DSN     image.file.dsn
```

UNL_IMAGECOPY_DATE

To override the Image Copy Date parameter for an unload program, specify:

```
UNL_IMAGECOPY_DATE     yyyy-mm-dd
```

UNL_IMAGECOPY_TIME

To override the Image Copy Time parameter for an unload program, specify:

```
UNL_IMAGECOPY_TIME     hh.mm.ss
```

UNL_IMAGECOPY_SELECT

To override the Image Copy Date Criteria parameter for an unload program, specify:

```
UNL_IMAGECOPY_SELECT     { A | B | L | S }
```

UNL_OBID

To process an Image Copy data set created on a different subsystem, specify:

```
UNL_OBID    [cid.]tblname  obid
```

cid.tblname
: The table name must be specified. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

obid The DB2 Object Identifier must be specified and is used to generate the OBID parameter or the ORIGINOBID parameter on the UNLOAD statement. See the appropriate BMC or IBM reference manual for more information.

COMMIT_COUNT

To override the commit count that was specified when the job was created, specify:

```
COMMIT_COUNT   value
```

The value can range from zero to the site limit.

COMMIT_MINUTES

To change commit processing from number of updates to elapsed time, specify:

```
COMMIT_MINUTES   value
```

The value is specified in minutes and will override the commit count. The value can range from 1 to 1440. The process report will be changed to reflect the change from the number of updates to elapsed time.

DEFCID

To override the default Creator ID in the Access Definition, specify:

```
DEFCID     cid
```

cid The default Creator ID to be used. This applies only to tables that are not explicitly qualified in the Access Definition.

Only one DEFCID parameter may be specified for an Extract Process. This override also affects the names of the tables in the relationships on the **Relationship Usage** list. If a relationship is not found for the updated table name, an error occurs when the extract is performed.

SEL

To override the selection criteria in the Access Definition for a table or to specify selection criteria for a table that does not have selection criteria in the Access Definition, specify:

```
SEL    [cid.]table  column  [ selcriteria ]
```

cid.table
The table name is required. If you do not specify the Creator ID (*cid*.), the default Creator ID defined in the Access Definition is assumed.

column The column name is required.

selcriteria
The selection criteria. This begins with an SQL operator. This specification is limited to 53 characters.

If you do not specify selection criteria, any selection criteria in the Access Definition is ignored for the current Extract Process.

Selection criteria can be specified for one or more columns in the table, but each criteria must apply to a different column.

You can specify only one SEL parameter for each column in a table. You can specify selection criteria overrides for as many columns in as many tables as you want as long as a separate SEL parameter is provided for each column.

SQL

To override the SQL WHERE Clause in the Access Definition for a table or to specify an SQL WHERE Clause for a table that does not have one in the Access Definition, specify:

```
SQL   [cid.]table  [ /correlation/ ]  [ where ]
```

cid.table
The table name must be specified. If you do not specify the Creator ID (*cid*.), the default Creator ID defined in the Access Definition is assumed.

/correlation/
Add or change a correlation name. If you specify a correlation name, it must immediately follow *table* and be enclosed in slashes.

where The SQL WHERE Clause. This must conform to the requirements specified for the Specify SQL WHERE Clause panel. However, the keyword WHERE is not required.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

If you do not specify an SQL WHERE Clause, any SQL WHERE Clause specified in the Access Definition is ignored for the current Extract Process.

You can specify the WHERE Clause override for more than one table as long as a separate SQL parameter is provided for each. You can specify only one SQL parameter for a table.

EVERY_NTH_ROW

To override the numeric value in the Access Definition used as a factor for selecting rows from a table, specify:

```
EVERY_NTH_ROW   [cid.]tblname   value
```

cid.tblname
Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

value A numeric value to specify a sampling factor for a table (*tblname*). Valid values are 1 through 65,535.

ROW_LIMIT

To override the numeric value in the Access Definition used to limit the number of rows selected from a table, specify:

```
ROW_LIMIT    [cid.]tblname   value
```

cid.tblname
Table name is required. If you do not specify the Creator ID (*cid*), the default Creator ID defined in the Access Definition is assumed.

value A numeric value to limit the number of rows selected from a table (*tblname*). Valid values are 1 through 4,294,967,295.

VAR

To override the default value of a substitution variable assigned in the Access Definition, specify:

```
VAR    varname   value
```

varname
The name of the substitution variable assigned in the Access Definition. A colon (:) in front of *VarName* is optional.

value The value for the substitution variable. You must enclose the value in single quotes if the variable is for a CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, BINARY, VARBINARY, DATE, TIME, TIMESTAMP, or TIMESTAMP WITH TIME ZONE column.

Note: If you specify a column name for the default value, do not enclose the value in quotes.

GROUP

To override the Group Selection Processing in the Access Definition for the Start Table or to specify Group Selection Processing when it has not been defined, specify:

```
GROUP    [column  VALUES=values  ROWS=rows]
```

column The name of the column in the Start Table for which Group Selection Processing is requested. The column must exist in the Start Table.

values The number of distinct values to select for the specified columns.

The value must be in the range 1 and 4,294,967,295. Specify an asterisk to obtain a specific number of rows from all distinct values of the column. (For additional information on Group Selection Processing, see Access Definitions in the *Common Elements Manual*.)

rows The number of rows to select for each value of the specified column.

The value must be in the range 1 and 4,294,967,295. Specify an asterisk to obtain all rows for each distinct value of the selected column.

Only one asterisk may be specified, therefore an asterisk cannot be specified for both values and rows.

This specification can be used with most selection criteria. However, the Every Nth specification is ignored.

Specify GROUP with no operands to ignore the Group Selection specifications in the Access Definition for the current Extract Process.

UNKNOWN

To override the default treatment for objects referred to in the Access Definition that no longer exist, specify:

```
UNKNOWN    { FAIL | ALLOW }
```

FAIL Terminate the Extract Process if any tables or relationships named in the Access Definition have become unknown since the batch process was initiated. The report contains a message noting the first unknown object. FAIL is the default.

ALLOW
Bypass the unknown tables and relationships and continue with the Extract Process. The report contains a message listing the unknown objects.

Since the Extract Process uses existing tables and the existing relationships between those tables to traverse the database, be aware that if an unknown table or relationship is bypassed, "related" tables in the Access Definition may not be included even when these tables exist.

UNKNOWN ALLOW is typically used when you override the default Creator ID with DEFCID. Changing the default Creator ID may result in naming tables that do not exist. The UNKNOWN ALLOW parameter lets you direct the Extract Process to skip these "unknown" tables.

WITH_UR

To override the extracting of uncommitted data from the database during the Extract Process, specify:

```
WITH_UR    { Y | N }
```

Use the Y operand to extract uncommitted data from the database. Use N to only archive committed data.

Note: This override is only available if the **Use Uncommitted Reads** site option is set to **U**.

Rules for Parameters

The following rules apply when specifying these parameters:

- One or more parameters may be specified in the JCL.
- Each parameter keyword must begin in the first space of the line.
- The qualifiers for the parameters must be space separated.
- If a parameter spans multiple lines, continue on the next line.
- You can comment the parameter list by specifying an asterisk, *, in the first position of each comment line.
- If multiple selection criteria are specified, they are ANDed or ORed depending on the type of Extract processing:
  - For batch execution of an Extract process generated online, criteria are combined according to your response on the Specify Selection Criteria panel.
  - For the Batch Utility Extract statement, criteria are combined according to the coloperator value of the SELECT keyword.
- If both an SQL WHERE Clause and selection criteria are specified for an individual table, the clauses are ANDed.
- You can store the parameters in a sequential file or a partitioned data set. The record length should be 80. If it exceeds 80, only the first 80 characters are processed. Sequence numbers are not allowed.

The parameters should conform to the same syntax required when specified directly in the jobstream.

Override Examples

To override the default Creator ID and specify selection criteria for two of the tables in the extract, insert in the JCL:

```
//PSDFOVRD DD *
DEFCID PSTDEMO2
* LIMIT SELECTION TO CUSTOMERS IN NEW JERSEY
* WHO HAVE ORDERS FOR WHICH THE
* FREIGHT CHARGES EXCEEDED $50.00
SEL CUSTOMERS STATE ='NJ'
SEL ORDERS FREIGHT_CHARGES >50.00
```

SEL requires at least one space between the column name and the selection criteria.

To override the SQL WHERE Clause for one table in the extract, insert in the JCL:

```
//PSDFOVRD DD *
* LIMIT SELECTION TO CUSTOMERS IN PRINCETON,
* NEW JERSEY
SQL PSTDEMO2.CUSTOMERS WHERE CITY=
'PRINCETON' AND STATE='NJ'
```

INSERT

Use an INSERT statement to insert or update rows in DB2 tables using data from an Extract File.

All options that may be specified using the online INSERT panel are also available using the batch INSERT processing statement. For an extract file on tape, you must use batch execution for an insert process.

```
INSERT
  EXTRACT_FILE ( { USEDD | explicitfilename  } )
  CONTROL_FILE ( File Allocation Parameters )
  { TABLE_MAP_DEFINE (parameters) ; | TABLE_MAP mapid.name
     [ PROCESS_MODE {INS | UPD | BOTH} ]
     [ DELETE_BEFORE_INSERT {TABLE | END} ]
     [ TMDEFCID defcreatorid ] }
  [ LOCK_TABLES {YES | NO} ]
  [ COMMIT_COUNT n ]
  [ COMMIT_MINUTES n ]
  [ DISCARD_COUNT n ]
  [ UNKNOWN {ALLOW | FAIL} ]
  [ AGING (parameters) ]
  [ REPORT_LEVEL {DETAIL | SUMMARY} ]
  [ RESTART {YES | NO} ]
  [ EMPTY_TABLE_FOUND_RC4 { YES | NO } ]
```

EXTRACT_FILE
Name of the Extract File to be used in the INSERT process.

USEDD
Use the file name specified in the PSDFEXTR control card statement.

explicitfilename
The fully qualified name of the Extract File.

CONTROL_FILE
Control File to be used in the INSERT process. Use the keywords shown in File Allocation Parameters to name and allocate the Control File.

TABLE_MAP
The name of the Table Map to be used. TABLE_MAP is required.

mapid.name
The fully qualified name of the Table Map.

TABLE_MAP_DEFINE
The Optim online process generates this keyword when it creates an INSERT job for batch execution.

When you create an INSERT job outside of the Optim online process, the best practice is to use the TABLE_MAP keyword to refer to a named Table Map in the Optim Directory.

Place the TABLE_MAP_DEFINE parameters within parentheses. A semicolon must follow the close parenthesis. (See TABLE_MAP_DEFINE Parameters for the allowable parameters.)

PROCESS_MODE
The process mode for any tables for which process mode is not specified in the Table Map.

INS Data rows are inserted.

UPD Data rows are updated.

BOTH Data rows are both inserted and updated (default).

DELETE_BEFORE_INSERT
Delete Before Insert option for tables for which delete before insert is not required by the Table Map. This keyword applies only to tables processed with Insert Mode. This keyword is valid only if the Site Option **Target Rows Del Before Isrt** is Y or I.

TABLE
Delete rows and commit after each table.

END Delete rows and commit at end.

LOCK_TABLES
Indicate whether to lock tables during processing. This keyword is relevant only if site options allow a user to lock tables.

YES Lock tables.

NO Do not lock tables (default).

COMMIT_COUNT
Number of rows processed between commit operations. If this keyword is omitted and the COMMIT_MINUTES keyword is omitted, the site limit is used.

n Specify a value from 1 to the site limit.

COMMIT_MINUTES
The number of minutes between commit operations. This keyword overrides a value specified with COMMIT_COUNT.

n Specify a number from 1 to 1440.

DISCARD_COUNT
The maximum number of discarded rows allowed. If this limit is met, the process is terminated. If you omit this keyword, there is no limit.

n Specify a value from 1 to 4,294,967,295.

UNKNOWN
Processing for tables referenced in the Table Map that are unknown.

ALLOW
Bypass table.

FAIL Terminate the INSERT (default).

TMDEFCID
Override for the default Creator ID specified in the Table Map.

defcreatorid
The Creator ID to be used.

AGING
This keyword indicates that date values in the source columns are to be aged. It provides parameters to be used in the aging process. See AGING Keyword Parameters for detailed information on using this keyword.

REPORT_LEVEL
The level of detail in the process report.

DETAIL
Produce a detailed report (default).

SUMMARY
Produce a summary report.

RESTART
Indicate whether this is a restart or retry attempt for an Insert process that failed earlier. Optim automatically determines whether to perform a restart or a retry.

YES Restart or retry processing.

NO Do not attempt to restart or retry processing (default).

EMPTY_TABLE_FOUND_RC4
Indicate the action taken and return code (RC) setting when empty tables have been detected during Insert batch processing.

YES Empty tables will be processed and a warning message will be issued with RC=4.

NO Empty tables will cause termination with an error and RC=12 (default).

Example

The following is an example of INSERT batch statement usage.

To insert the contents of the Extract File PSTUSER.DISC, use INSERT ONLY processing, and delete all rows from the tables before inserting the new rows, specify:

```
INSERT EXTRACT_FILE PSTUSER.DISC
   CONTROL_FILE  (DSNAME PSTUSER.INSCTRL)
   TABLE_MAP  PSTUSER.MAP9
   PROCESS_MODE  INS
   DELETE_BEFORE_INSERT  TABLE
```

INSERT Override Keywords

When you submit the job, you can override the default destination Creator ID defined in the Table Map used for the Insert Process. This is especially convenient when you want to apply different Extract Files to a single set of tables or a single Extract File to multiple sets of tables using common Insert Process JCL.

Any Extract File can be used in an Insert Process as long as at least one table name on the file matches one table name on the Table Map. The Creator IDs do not have to match. If any table does not match on the Table Map, it is not included in the process.

Use the PSDFOVRD DD statement in the JCL to provide the desired overrides.

COMMIT_COUNT

To override the commit count that was specified when the job was created, specify:

```
COMMIT_COUNT   value
```

The value can range from zero to the site limit.

COMMIT_MINUTES

To change commit processing from number of updates to elapsed time, specify:

```
COMMIT_MINUTES   value
```

The value is specified in minutes and will override the commit count. The value can range from 1 to 1440. The process report will reflect the change from the number of updates to elapsed time.

Creator ID

To override the default destination Creator ID specified on the Table Map specify:

```
DEFCID   cid
```

where *cid* is the default Creator ID to be used. This applies only to destination tables that are not explicitly qualified in the Table Map. Only one DEFCID parameter may be specified for an Insert Process.

Date Aging

To override the date aging specifications for the Insert Process, specify one or more of the following:

AGE_AMT_YEAR
For incremental aging, the amount to increment by years.

AGE_AMT_MONTH
For incremental aging, the amount to increment by months.

AGE_AMT_WEEK
For incremental aging, the amount to increment by weeks.

AGE_AMT_DAY
For incremental aging, the amount to increment by days.

About AGE Parameters

The value must be numeric. Specify whether the value increments (+) or decrements (-). Increment is the default. For example, +15 or 15 increments the dates and -15 decrements the dates.

The combination of **AGE_AMT** parameters specifies the aging.

- To specify an explicit date, specify a value for YEAR that is greater than or equal to 1582. Then, values must be specified for MONTH and DAY to define an explicit date. (A value for WEEK is invalid for an explicit date.)
- To specify a date other than an explicit date, you can specify values for any or all units. The valid ranges for the values are:

```
YEAR    -2500 to +1581
MONTH   -30000 to +30000
WEEK    -30000 to +30000
DAY     -99999 to +99999
```

AGE_AMT_RULE
For incremental aging, the amount of occurrences of an aging rule to increment. For example, assume the specified rule is NEXTPAYDAY, specify 4 to age to the fourth payday from the current date in the column.

AGE_BASE_DATE
Specify a base date used by Move to determine the difference between this value and the AGE_TARGET_DATE. The difference is used as the number of days to age the data.

This must be specified in the format: yyyy/mm/dd or yyyy/ddd.

The current date is the default.

AGE_TARGET_DATE

Specify the target date used by Move to determine the difference between this value and the AGE_BASE_DATE. The difference is used as the number of days to age the data.

This must be specified in the format: yyyy/mm/dd or yyyy/ddd.

This is required if AGE_BASE_DATE is specified.

TABLE

Specifies the name of the aging rule table to be used. Specify the name of a valid partitioned data set member.

AGE_RULE

Indicates the default aging rule to be used for any date not explicitly assigned an aging rule. Specify any value in the aging rule table specified for the TABLE statement. (For more information about the aging rule table, see Customize Aging Rules in the *Customization Guide*.)

If not specified, no aging rule is applied.

PIVOT_YEAR

Indicates which century to apply to two⌂digit year values. This information is used for aging rules and to provide the century in the output if so formatted.

Specify a two-digit value from 00 through 99. This value determines the threshold. If not specified, a pivot year is not used.

For example, assume the value is 65. All two-digit years 65 or over are assumed to be in the 20th century (19xx); all two-digit years that are less than 65 are assumed to be in the 21st century (20xx).

LIST_INVALID

Specifies whether details of the invalid dates encountered during the aging are listed at the beginning of the process report.

Specify Y to list the details or N to not list them. The default is Y.

LIST_SKIPPED

Specifies whether details of the skipped dates encountered during the aging are listed at the beginning of the process report. Dates are skipped when the date cannot be aged because the value is not a valid date but has special meaning to the application.

For example, “000000” or “999999” are not valid dates but may be special indicators for the application. Other examples of skipped dates are those containing only spaces, hex zeroes, or hex “FF”.

Specify Y to list the details or N to not list them. The default is Y.

PUT_INVALID

Specifies whether the rows with invalid dates are written to the database.

Specify Y to write the rows or N to not write them. The default is Y.

PUT_SKIPPED

Specifies whether the rows with skipped dates are written to the database.

Specify Y to write the records or N to not write them. The default is Y.

PROCESS_DATE_COLUMNS

Specifies whether date columns not explicitly mapped are to be aged.

A All DATE and TIMESTAMP columns and the columns mapped to AGE are aged.

U Only columns mapped to AGE are aged.

UNKNOWN

To ignore any tables referred to in the Table Map that do not exist when performing the Insert Process, specify:

```
UNKNOWN { FAIL | ALLOW }
```

FAIL Terminate the insert if any tables or relationships named in the Table Map are unknown. The report will contain a message listing the first unknown object. This is the default.

ALLOW
Bypass the unknown tables and relationships and continue with the Insert Process. The report will contain a message listing the unknown objects.

UNKNOWN ALLOW is most frequently used when you override the default Creator ID with DEFCID. Changing the default Creator ID may result in naming destination tables that do not exist. The UNKNOWN ALLOW parameter enables you to direct the Insert Process to skip these “unknown” tables.

Store Overrides

You can store these parameters in a sequential file or a partitioned data set rather than specify them directly in the jobstream. However, these parameters must be the only data in the file. (You cannot use the same file used for a batch Extract Process if selection criteria, SQL WHERE Clause, or Group Selection Processing parameters are also included. Also you cannot use this file for an Extract Process if the YEAR parameter is included.)

LOAD

You can use the LOAD statement to transform the contents of an Extract or Archive File to load utility format. You can then use the files created by this statement as input to a load utility to load the data into a database.

```
LOAD
  INPUT_FILE { ( USEDD ) | explicitfilename }
  CONTROL_FILE ( File Allocation Parameters )
  TABLE_MAP mapid.name
  LOAD_DSNPRFX dsnprefix
     [ LOAD_BLKSIZE nnnnn ]
     [ LOAD_UNIT { SYSDA | unitname } ]
     [ LOAD_DSNTYPE {BASIC | LARGE | EXTREQ | EXTPREF} ]
     [ LOAD_VOLCOUNT {1 | nn} ]
     [ LOAD_PRIMARY n{ C | T | K | M } ]
     [ LOAD_SECONDARY n{ C | T | K | M } ]
     [ LOAD_MGMTCLASS mgmtclass ]
     [ LOAD_STORCLASS storclass ]
  [ CLEAR_ALL_TABLES { YES | NO } ]
  [ SORT_ROWS { YES | NO } ]
  [ AGING (parameters) ]
  [ REPORT_LEVEL { DETAIL | SUMMARY } ]
  [ FORCE_EXTERNAL_LOBS { YES | NO } ]
  [ ENFORCE_RI { YES | NO } ]
  [ EMPTY_TABLE_FOUND_RC4 { YES | NO } ]
```

INPUT_FILE
Identify the name of the extract or archive file you want to use in the LOAD process. Specify:

USEDD
Use the file name specified in the PSDFEXTR control card statement.

explicitfilename
The fully qualified name of the extract or archive file.

CONTROL_FILE
Identify the control file you want to use in the LOAD process. Use the keywords shown in File Allocation Parameters to name and allocate the control file.

TABLE_MAP
Identify the name of the table map you want to use. TABLE_MAP is required.

mapid.name
The fully qualified name of the table map.

LOAD_DSNPRFX
If appropriate, type the data set name prefix for all Loader and Field Specification files generated by the LOAD statement. The specified prefix can consist of up to 35 characters and must follow standard data set naming conventions. This parameter allows the Optim solution to create the required Field Specification files dynamically during Load preparation step execution, instead of statically during JCL generation.

When you specify this parameter, the Field Specification statements for all loaded tables and their associated data are stored in dynamically allocated data sets, which use the following naming convention.

dsnprefix
Specify a prefix up to 35 characters, using standard data set naming conventions. The LOAD statement names the output files as follows:

dsnprefix.L*seq* for Loader files

dsnprefix.X*seq* for Field Specification files

where *dsnprefix* is the DSN prefix specified using this parameter, and *seq* is a sequential number assigned by the Optim solution.

LOAD_BLKSIZE
Specify the block size for allocating Loader files. The default value is 23476.

nnnnn Specify a block size from 5000 to 32767.

LOAD_UNIT
Specify the unit type for allocating Loader files. The default value is SYSDA.

unitname
Specify a site-defined unit type.

LOAD_DSNTYPE
Specify the data set name type, as defined by the Data Facility Storage Management Subsystem (DFSMS) allocation documentation. The Optim solution supports both extended and large sequential data sets for the loader data set. Specify:

BASIC
An extended or large format sequential data set need not be allocated (default).

LARGE
A large format sequential data set is to be allocated.

EXTREQ
An extended data set is required.

EXTPREF
An extended data set is preferred.

LOAD_VOLCOUNT
Specify the number of volumes that can be assigned to the file. The default on the panel is 1. If the default value is deleted and not replaced, the volume count for the input data set is used. If needed, the Optim solution will automatically increase the specified volume count, up to the maximum value allowed by DFSMS.

LOAD_PRIMARY

Specify the primary quantity of space you want allocated in a given space unit. If omitted, a default space allocation is calculated. Specify:

n The space to allocate. Specify any integer up to the maximum value allowed by DFSMS for the selected space unit.

C, T, K, or M

The primary space unit for allocation purposes must be one of the following:

- **C** for cylinders
- **T** for tracks
- **K** for kilobytes
- **M** for megabytes

For example, an entry of 25M would mean allocate 25 megabytes.

LOAD_SECONDARY

Specify the secondary quantity of space you want allocated in a given space unit. If omitted, a default space allocation is calculated. Specify:

n The space to allocate. Specify any integer up to the maximum value allowed by DFSMS for the selected space unit.

C, T, K, or M

The secondary space unit for allocation purposes must be one of the following:

- **C** for cylinders
- **T** for tracks
- **K** for kilobytes
- **M** for megabytes

LOAD_MGMTCLASS

Identify the management class used to obtain data management information to allocate the data set. If omitted, the installation default is used.

mgmtclass

The management class, as defined by your system administration. The management class is used to obtain the data management-related information (migration, backup, and retention criteria) for the allocation of the data set.

LOAD_STORCLASS

Identify the storage class used to obtain storage information to allocate the data set. If omitted, the installation default is used.

storclass

The storage class, as defined by your system administration. The storage class is used to obtain the storage-related information for the allocation of the data set.

CLEAR_ALL_TABLES

Indicate whether you want load utility processing to stop or continue if the utility states that a table space is to be cleared before data is loaded. Specify:

YES Clear the table space and continue processing. Data might be lost if all tables in the table space are not reloaded.

NO Stop processing and do not clear the table space (the default value).

SORT_ROWS

Indicate whether you want to sort rows with a cluster index if one exists. Specify:

YES Sort rows.

NO Do not sort rows (the default value).

AGING

This keyword indicates that date values in the source columns are to be aged. It provides parameters to be used in the aging process. See AGING Keyword Parameters for detailed information about using this keyword.

REPORT_LEVEL

Indicate the level of detail you want in the process report. Specify:

DETAIL

Produce a detailed report (the default value).

SUMMARY

Produce a summary report.

FORCE_EXTERNAL_LOBS

Indicate whether Large Object (LOB) data from the Archive or Extract File is always to be stored in a Partitioned Data Set - Extended (PDS/E) data set.

When LOB data is stored in a PDS/E data set, the Load utility input file (SYSREC) will not contain LOB data. Instead, the input file will contain reference pointers that point to the PDS/E data set and member name in which each instance of LOB data is stored.

Storing LOB data in a PDS/E data set reduces the length of table rows in the input file. It also allows the Load utility to load table rows that would otherwise be too long to load. Specify:

YES Always store LOB data from the archive or extract file in a PDS/E data set that is separate from the input file.

NO Store LOB data from the archive or extract file in a PDS/E data set only when the input file would otherwise contain table rows greater than 32 KB. NO is the default value.

ENFORCE_RI

Indicate whether you want Referential Integrity (RI) constraints enforced during the Load process. Specify:

YES Check RI and discard any invalid row encountered during the Load, and include the ENFORCE YES clause in the LOAD DATA statement.

NO Suspend RI checks during the Load and leave the table in Check Pending status, and include the ENFORCE NO clause in the LOAD DATA statement. Specify NO if you want to load all rows regardless of referential integrity and check RI after the load is done. NO is the default value.

EMPTY_TABLE_FOUND_RC4

Indicate the action you want taken when empty tables are detected during Load batch processing. Specify:

YES Process the empty tables, issue a warning message, and end processing with a return code of RC=4.

NO Do not process the empty tables and end processing with a return code of RC=0. NO is the default value.

Example

Use the following statement to transform the contents of the DEPT14.TRADES file to Load utility format, using the DEPT14.TRADESMAP table map.

```
LOAD

   INPUT_FILE   DEPT14.TRADES
   CONTROL_FILE (DSNAME  DEPT14.CTRLFILE  MODE  REP)
   TABLE_MAP  DEPT14.TRADESMAP
   LOAD_DSNPRFX PSTJS.LOADOUT
```

Required Load Utility DD Statements

You must include the appropriate DD statements for the load utility program you are using. See the documentation for the program for information about the required DD statements.

If you do not specify the LOAD_DSNPRFX keyword in the LOAD batch utility, you must include a PSDFL*nnn* DD statement for each table space into which you are loading data. For example,

```
//PSDFL001 DD DSN=PSTJS.SUPDB.PSTJS.STEP001,
//    DCB=(DSORG=PS,RECFM=VB,LRECL=23472,BLKSIZE=0),
//         DISP=(MOD,CATLG,CATLG),
//         UNIT=SYSDA,
//         SPACE=(0,1,1),RLSE,,ROUND)
```

Also, for each table space into which you are loading data, you must include the following statements in the load utility JCL:

```
//SYSREC   DD DSN=dsnprefix.Lseq,DISP=OLD
//SYSIN    DD DSN=load.data.ds,DISP=OLD
//         DD DSN=dsnprefix.Xseq,DISP=OLD
```

The LOAD batch utility statement generates the *dsnprefix*.L*seq* (Loader) and *dsnprefix*.X*seq* (Field Specification) output files. For example, the following LOAD output summary lists the Loader and Field Specification output files:

```
CONVERSION OF ARCHIVE OR EXTRACT FILES IS COMPLETE.
  SUMMARY OF RESULTS BY TABLE SPACE:
   TABLE SPACE PSTJS
      LOAD OUTPUT FILE (PSDFL001): PSTJS.LOADOUT.L001
      FIELD-SPEC FILE (PSDFX001): PSTJS.LOADOUT.X001
```

The *load.data.ds* data set contains the control statement for the load utility you are using. For example, the DB2 LOAD control statement includes the following keywords:

```
LOAD DATA REPLACE
  LOG YES
  SORTKEYS 0
  ENFORCE NO
  DISCARDS 0
```

By comparison, if you were to use the online screens, Optim would generate the following DB2 LOAD control statements:

```
//DSNUPROC.SYSIN     DD *
 LOAD DATA REPLACE
   LOG YES
   SORTKEYS 0
   ENFORCE NO
   DISCARDS 0 CONTINUEIF(1:1)=X'FF'
   INTO TABLE PSTDEMO.SALES WHEN (1:2) = X'0001'
     (SALESMAN_ID POSITION (3:8) CHAR,
      SALESMAN_NAME POSITION (9:28) CHAR,
      AGE POSITION (29:32) INTEGER,
      SEX POSITION (33:33) CHAR,
      TERRITORY POSITION (34) VARCHAR,
      MANAGER_ID POSITION (52:57) CHAR
      NULLIF (50:50) = X'FF')
```

Large Object (LOB) data from the archive or extract file must be stored either inline in the Load Process input file (SYSREC) or externally in a Partitioned Data Set - Extended (PDS/E) data set. Do not change the generated JCL to store LOB data in a Partitioned Data Set (PDS) data set. Also, ensure that the PDS/E data set used in the generated JCL is empty. If the PDS/E data set is not empty, the members created in this step might replace members in the existing data set.

CENTERA_OPTIONS

Use the CENTERA_OPTIONS statement to provide options for using a Centera Server with Archive File processing.

This statement applies to all subsequent Batch Utility control statements until another CENTERA_OPTIONS statement is encountered. This statement overrides options specified in the Centera Site Options panel.

```
CENTERA_OPTIONS
  [ POOL  poolname ]
  [ DELETE_MVSFILE  { YES | NO } ]
  [ RETENTION_MODE { DEFAULT | NONE | INFINITE
     | INTERVAL  [ YEARS  n ] [DAYS  n ] } ]
  [ RECALL { ABORT | AUTO } ]
  [ STAGE  { YES | NO } ]
```

Copy Keywords

Use the following keywords to specify options for copying an Archive File to a Centera Server.

POOL The Centera Pool to which the Archive File is copied. This keyword is required if more than one pool is defined in the Centera Site Options panel. If only one pool is defined, this keyword is optional.

poolname

The Centera Pool Name, defined in Centera Site Options, to which the Archive File is copied.

DELETE_MVSFILE

Indicator to delete the Archive File from disk after the file is copied to the Centera Server.

YES Automatically delete the Archive File stored on disk.

NO Do not delete the Archive File from disk (default).

RETENTION_MODE

Indicator for the period for which the Archive File is retained on the Centera Server before it can be deleted. This keyword is allowed only if the Centera Site Option **Minimum Retention** is set to User.

DEFAULT

Use the default retention period, based on the Centera configuration.

NONE

No retention period is used (default). The Archive File can be deleted from Centera at any time.

INFINITE

Retain the Archive File on Centera forever; the file cannot be deleted.

Note: Any Archive File assigned a retention period over 9,999 days or 27 years is considered permanent. A permanent retention period can be deleted or overwritten by using the ALTER statement of the IBM Utility program, IDCAMS, to reduce the retention period.

INTERVAL

Protect the Archive File from deletion for a specified period. You can specify a number of years, days, or a combination of both.

YEARS

If **Interval** is selected, enter the number of years (0-100) to protect an Archive File from deletion.

DAYS If **Interval** is selected, enter the number of days (0-18,300) to protect an Archive File from deletion.

Note: The combined total of years and days must be greater than 0.

Recall Keywords

Use the following keywords to provide options for recalling an Archive File from a Centera Server.

RECALL

Indicator for recalling Archive Files from a Centera Server. This parameter is allowed only if the Centera Site Option **Recall Processing** is set to USER or PROMPT.

ABORT

If an Archive File must be recalled for processing, the process is terminated with an error message.

AUTO

An Archive File on Centera is recalled when needed (default).

STAGE

Indicator for method of recall.

YES Recall the Archive File in 2 MB stages (default).

NO Recall the Archive File in its entirety.

Example

Specify the following to copy an Archive File to the Centera Pool named CENT1, delete the file from disk after it is copied, and retain the file on the Centera Server for 15 years.

```
CENTERA_OPTIONS
  POOL  CENT1
  DELETE_MVSFILE  YES
  RETENTION_MODE  INTERVAL
     YEARS  15
```

TIVOLI _OPTIONS

Use the TIVOLI_OPTIONS statement to provide options for using the Tivoli Storage Manager Server with Archive File processing.

This statement applies to all subsequent Batch Utility control statements until another TIVOLI_OPTIONS statement is encountered. This statement overrides options specified on the Tivoli Site Options panel.

```
TIVOLI_OPTIONS
  [ ALIAS  name ]
  [ FILE_SPACE prefix ]
  [ MGMT_CLASS name ]
  [ DELETE_MVSFILE  { YES | NO } ]
  [ RECALL { ABORT | AUTO } ]
```

Copy Keywords

Use the following keywords to specify options for copying an Archive File to Tivoli.

ALIAS

The Tivoli alias definition to be used for the copy operation. This keyword is required if more than one alias is defined on the Tivoli Site Options panel. If only one alias is defined, this keyword is optional.

name The Tivoli alias definition to be used for the copy operation.

FILE_SPACE
Name of the filespace prefix to which the Archive File is copied. If this keyword is omitted, the filespace specified in the alias is used.

prefix Name of the filespace to which the Archive File is copied.

MGMT_CLASS
Name of the management class to be used. If this keyword is omitted, the management class specified in the alias is used.

name Name of the management class.

DELETE_MVSFILE
Indicator to delete the Archive File from disk after the file is copied to Tivoli.

YES Automatically delete the Archive File stored on disk.

NO Do not delete the Archive File from disk (default).

Recall Keyword

RECALL
Indicator for recalling Archive Files from Tivoli. This parameter is allowed only if the Tivoli Site Option **Recall Processing** is set to USER or PROMPT.

ABORT
If an Archive File must be recalled for processing, the process is terminated with an error message.

AUTO
An Archive File on Tivoli is recalled when needed (default).

Example

Specify the following to save Archive files on Tivoli using the alias TIVOL3, retain the Archive files on disk after they are stored on Tivoli and terminate processing if an attempt is made to recall the file from Tivoli:

```
TIVOLI_OPTIONS ALIAS TIVOL3 RECALL ABORT
```

IGNORE_GENERIC_RELS

An IGNORE_GENERIC_RELS statement allows you to specify whether generic relationships are ignored in process requests.

```
IGNORE_GENERIC_RELS { ALWAYS | NEVER }
```

Keywords for an IGNORE_GENERIC_RELS statement are

ALWAYS
Always ignore generic relationships during processing.

NEVER
Never ignore generic relationships during processing.

This statement is valid only when the Site Option **Ignore Generic Rels** is set to User. If omitted, generic relationships are used in process requests.

Chapter 5. File Allocation Parameters

During batch utility processing, one or more output files may be created depending upon the batch statement and options you use.

To create an output file, specify the file name and, optionally, parameters for attributes such as block size and processing mode. Usually only a few parameters are necessary to allocate a file, as shown in the following examples:

To create a new archive file on disk, allowing Optim to compute the space needed, and using site defaults for unit and volume information, specify:

```
ARCHIVE_FILE (DSNAME FOPUSER.NJCUST04)
```

The next example creates an external file, with a specified block size, volume assignment and space requirement. If the file name already exists, it will be deleted and reallocated.

```
EXTERNAL_FILE (DSNAME FOPUSER.CUSTOMER.NEWYORK
               MODE REP VOLUME FOP003 BLKSIZE 8000
               PRIMARY 100T SECONDARY 20T)
```

The parameters in this section should be used in any batch statement syntax where the words “File Allocation Parameters” appear.

```
type_FILE( { USEDD |DSNAME dsname }
   [ MODE { REP | STOP | USE | GEND | GENT } ]
   [ TAPE  { YES | NO } FILENBR [ n ] ]
   [ MGMTCLASS mgmtclass ]
   [ STORCLASS storclass ]
   [ DATACLASS dataclass ]
   [ DSNTYPE {BASIC | LARGE | EXTREQ | EXTPREF} ]
   [ VOLUME volser ]
   [ VOLCOUNT n ]
   [ UNIT unit ]
   [ PRIMARY n { C | T | K | M } ]
   [ SECONDARY n { C | T | K | M } ]
   [ BLKSIZE nnnn ] } )
```

Parameters must be enclosed in parentheses and may be in any order. DSNAME is always required. Multiple parameters can be specified on a single line.

File Type parameter

*type*_**FILE**

The file type parameter is required in the Archive, Extract, Deferred_Delete, Insert, Convert or Compare statement and must precede any additional parameters. Valid values are:

- ARCHIVE_FILE
- DUPE_ARCHIVE_FILE
- INDEX_FILE
- DUPE_INDEX_FILE
- COMPARE_FILE
- CONTROL_FILE
- CONVERTED_FILE
- EXTRACT_FILE
- EXTERNAL_FILE
- SUBSET_FILE

USEDD

Indicates that a DD statement is present in the JCL to define the file. If you use the USEDD parameter, you cannot specify any other file allocation parameters.

The ddnames are:

PSDFEXTR for archive and extract files

PSDFCTRL for control files

PSDFIDX for archive index files

PSDFASUB for subset files

DSNAME

Fully qualified name of the file. You may specify the name of a new or existing file.

dsname

The name of the file. If the file exists, you must use MODE REP or MODE USE.

MODE

Processing mode for the file. Allowable values are:

REP Allocate the file unconditionally. If the file exists, delete and reallocate it.

STOP Stop processing and display an error message if the file exists (default).

USE Use the existing file. Data in the existing file will be overwritten. If the file does not exist, stop processing and display an error message.

GEND (for archive files only)

Generate the file name by appending the suffix ".D*yyymmdd*" to the DSNAME value, where *yyy* is the last 3 bytes of the year. If this option is used, the maximum size of the DSNAME parameter is 32 characters.

Note: If you use this parameter and a file already exists containing the name to be suffixed, the existing file will not be deleted or renamed to include the suffix and may consequently be orphaned.

GENT (for archive files only)

Generate the file name by appending the suffix ".D*yymmdd*.H*hhmm*" to the DSNAME value. If this option is used, the maximum size of the DSNAME parameter is 26 characters.

Note: If you use this parameter and a file already exists containing the name to be suffixed, the existing file will not be deleted or renamed to include the suffix and may consequently be orphaned.

TAPE Indicator for tape file. This setting applies to archive files, any duplicate archive files, and extract files.

YES Assign the file to tape.

NO Assign the file to disk (default).

FILENBR

The sequence of the file on the tape. If omitted, the default value is 1.

n File sequence number.

MGMTCLASS

The management class used to obtain data management information to allocate the data set. If omitted, the installation default is used.

class The management class.

STORCLASS

The storage class used to obtain storage information to allocate the data set. If omitted, the installation default is used.

class The storage class.

DATACLASS

The data class used to obtain information to allocate the data set. If omitted, the installation default is used.

class The data class.

DSNTYPE

The data set name type, as defined by the Data Facility Storage Management Subsystem (DFSMS) allocation documentation. Optim supports both extended and large sequential data sets. Specify:

BASIC

An extended or large format sequential data set need not be allocated (default).

LARGE

A large format sequential data set is to be allocated.

EXTREQ

An extended data set is required.

EXTPREF

An extended data set is preferred.

VOLUME

The serial numbers of the volumes on which the file is to reside. A maximum of six volsers are allowed. If more than one volser is specified, the values must be enclosed in parentheses. If this parameter is omitted, the operating system assigns a value.

volser The volume serial number.

VOLCOUNT

The maximum number of volumes that may be assigned to this file. If you specify this parameter, omit the VOLUME parameter. If you omit VOLCOUNT, the VOLUME parameter controls the maximum number of volumes assigned.

nn The maximum number of volumes.

UNIT The UNIT parameter used to allocate the data set. If omitted, the site default is used. Omit this parameter for an index file.

unit The unit type.

PRIMARY

The primary quantity of space to allocate in the indicated space units. If omitted, Optim computes a default space allocation.

Note: For an archive or extract file, any unused space remaining after the job completes is automatically released.

n The space allocation followed by the space unit. Valid space unit values are:

C	for Cylinders
T	for Tracks
K	for Kilobytes
M	for Megabytes

SECONDARY

The secondary quantity of space to allocate in the indicated space units. If omitted, Optim computes a default space allocation. (The SECONDARY parameter has the same operands as the PRIMARY parameter.)

BLKSIZE

Specifies the block size to be used for the file. If omitted, Optim will calculate an appropriate block size. For an archive file or an extract file, the minimum value is 7944.

nnnn The block size.

Allocating Image Copy Data Sets

To archive, extract or compare data from DB2 image copy data sets in multiple partitions of the same tablespace stored on the same tape volume, you must manually edit the JCL to allocate the data sets. Multiple image copy data sets cataloged on the same tape volume can not be allocated using dynamic allocation. This is a z/OS limitation. If you attempt to use dynamic allocation, the archive, extract, or compare process fails with a dynamic allocation error.

1. If you are using Optim v7.1 or earlier, apply the provided fix: http://www.ibm.com/support/docview.wss?uid=isg1PM52820#more

   Fixed component name: OPTIM MOVE FOR

   Fixed component ID: 5655V0700

   Applicable component levels: R710 PSY UK78105 and UP12/04/21 P F204
2. For image copy data sets of one or more partitioned tablespaces on the same tape, either:
   - concatenate image copy data sets under one DD Name of your choice, or
   - code individual DD Names for allocation of image copy data sets. Use a pattern, such as *xxxxPPPP* where *xxxx* is the DD Name prefix and *PPPP* is the maximum number of partitions in that tablespace. For example, if a tablespace contains 5 partitions, the DD Name would be IMGC0001, IMGC0002, and so on.

Example:

Table	**Partition**	**Tape Allocation Requirement**
SALES	TAPEVL1 - LABEL(1,SL)	// T01$0001 DD DISP=OLD,DSN=(dsname) IC Dataset for SALES Table
CUSTOMER	PART1 TAPEVL1 - LABEL(2,SL)	// T02$0001 DD DISP=OLD,DSN=(dsname) IC Dataset for CUSTOMER Table Part #1
CUSTOMER	PART2 TAPEVL1 - LABEL(3,SL)	// T02$0002 DD DISP=OLD,DSN=(dsname) IC Dataset for CUSTOMER Table Part #2
CUSTOMER	PART3 TAPEVL1 - LABEL(4,SL)	// T02$0003 DD DISP=OLD,DSN=(dsname) IC Dataset for CUSTOMER Table Part #3
CUSTOMER	PART4 TAPEVL1 - LABEL(5,SL)	// T02$0004 DD DISP=OLD,DSN=(dsname) IC Dataset for CUSTOMER Table Part #4

Table	Partition	Tape Allocation Requirement

CUSTOMER	PART5 TAPEVL1 - LABEL(6,SL)	// T02$0005 DD
		DISP=OLD,DSN=(dsname)
		IC Dataset for CUSTOMER Table Part #5

ORDERS	TAPEVL1 - LABEL(7,SL)	// T03$0001 DD
		DISP=OLD,DSN=(dsname)
		IC Dataset for ORDERS Table

DETAILS	TAPEVL2 - LABEL(1,SL)	JCL allocation not required because it resides on a different tape

Chapter 6. AGING Keyword Parameters

The AGING keyword specifies that date values in the source columns are to be aged. It provides the parameters to be used in the aging process.

Note: You must specify a value for only one aging method: Explicit, Incremental, or Target. For Incremental aging, the combined values of Years, Months, Weeks, and Days cannot result in a year value greater than 3999. If the value exceeds 3999, an error occurs in processing.

The following AGING keyword parameters must be contained in parentheses and separated by spaces.

```
AGING( {  EXP_DATE date |
<  YEARS n | MONTHS n | WEEKS n |  DAYS n > |
 BUSINESS_UNITS n |
 BASE_DATE date TARGET_DATE date }
[ DEF_RULE_TABLE tablename  DEF_RULE_NAME rulename ]
[ PIVOT_YEAR nn ]  [ PROCESS_DATE_COLUMNS { A | U } ]
[ REPORT_INVALID { YES | NO } ] [ REPORT_SKIPPED { YES | NO } ]
[ OUTPUT_INVALID { YES | NO } ] [ OUTPUT_SKIPPED { YES | NO } ] )
```

Explicit Parameter

EXP_DATE

Specify an explicit date for aging. The date must be in the form YYYY/MM/DD or YYYY/DDD (a Julian date). The aging rule is applied to this date.

Incremental Parameters

YEARS

Adjust the date by *n* number of years.

+nnnnn | *-nnnnn*

Increment or decrement the value in the column by the number of years specified. Specify one to four digits in the following range: -2500 to +1581. A plus sign or a minus sign preceding the value indicates whether the date is to be incriminated or decremented. Increment is the default.

MONTHS

Adjust the date by *n* number of months.

+nnnnn | *-nnnnn*

Increment or decrement the value in the column by the number of months specified. Specify one to five digits in the following range, -30000 to +30000. A plus sign or a minus sign preceding the value indicates whether the date is to be incriminated or decremented. Increment is the default.

WEEKS

Adjust the date by *n* number of weeks.

+nnnnn | *-nnnnn*

Increment or decrement the value in the column by the number of weeks specified. Specify one to five digits in the following range, -30000 to +30000. A plus sign or a minus sign preceding the value indicates whether the date is to be incriminated or decremented. Increment is the default.

DAYS Adjust the date by *n* number of days.

+nnnnn | *-nnnnn*

Increment or decrement the value in the column by the number of days specified. Specify one to five digits, in the following range -99999 to +99999. A plus sign or a minus sign preceding the value indicates whether the date is to be incriminated or decremented. Increment is the default.

BUSINESS_UNITS

Specify the date adjustment by business rule units.

nnnnn Adjust the value in the column by *n* number of occurrences of the specified business rule unit. Specify one to five digits in the range, 0 to 30000. (Incriminating and decrementing is controlled by the rule.) For example, if the Aging rule is specified as NEXTPAYDAY, the date is adjusted by the specified number of paydays. Therefore, a 4 in BUSINESS_UNITS adjusts the date to the fourth payday after the date value in the column.

Target Parameters

BASE_DATE

Specify an explicit date as the origination or starting date for calculating the aging amount. If you omit BASE_DATE, the current date is assumed.

date Must be in the form YYYY/MM/DD or YYYY/DDD (a Julian date).

TARGET_DATE

Specify an explicit date as the target for calculating the aging amount. The aging amount is determined by the difference between the Base Date and the Target Date.

date Must be in the form YYYY/MM/DD or YYYY/DDD (a Julian date).

Other Parameters

DEF_RULE_TABLE

The name of the aging rule table to be used. (The default is the site default aging rule table.)

DEF_RULE_NAME

The default aging rule for any date column not explicitly assigned an aging rule. This must be a value in the default aging rule table or blank.

If DEF_RULE_NAME is blank, no aging rule is applied to any aged data that is not explicitly assigned a rule. (For details about aging rule tables, see Customize Aging Rules in the *Customization Guide*.)

PIVOT_YEAR

The year used to determine the century to assign to two-digit year values. For example, if the PIVOT_YEAR value is 65, all two-digit years that are 65 or greater are assumed to be in the 20th century (19xx) and all two-digit years that are less than 65 are assumed to be in the 21st century (20xx). This information is necessary to properly age the data.

Note: If the Column Map specifies a Pivot Year value for an individual column, the PIVOT_YEAR value is ignored for that column.

nn Specify a two-digit value from 00 to 99.

PROCESS_DATE_COLS

The type of date columns to be aged. Specify:

A Age all DATE and TIMESTAMP columns and the columns mapped to AGE.

U Age only columns mapped to AGE.

If a Column Map is not defined or the AGE function is not specified on the selected Column Map, date values apply only to DATE and TIMESTAMP columns.

REPORT_INVALID
Indicate whether to list details of invalid dates encountered during the process in the process report. Specify:

YES Report invalid dates.

NO Do not report invalid dates.

REPORT_SKIPPED
Indicate whether to list details of skipped dates encountered during the process in the process report. (Dates are skipped when the value is not a valid date but has special meaning to the application. See "Skipped Columns.") Specify:

YES Report skipped dates.

NO Do not report skipped dates.

OUTPUT_INVALID
Indicate whether rows with invalid dates are written to the destination database. Specify:

YES Write rows with invalid dates to the database.

NO Do not write rows with invalid dates to the database.

OUTPUT_SKIPPED
Indicate whether the rows with skipped dates are written to the database. (See "Skipped Columns.") Specify:

YES Write rows with skipped dates to the database.

NO Do not write rows with skipped dates to the database.

Skipped Columns

Frequently, values that are not valid dates are inserted into date columns to indicate special handling or conditions. Rather than treat these non-date values as invalid or errors, the process "skips" them. That is, when a column contains such a value, the column is bypassed. Since there is no error, processing continues with the next date column.

The user can indicate whether skipped values are noted in the Aging Report and if rows with skipped dates are written to the output file.

To determine whether a column should be skipped, the process evaluates the column value.

- If the column contains all spaces, hex zeros (low-values) or hex "FF" (high-values), it is skipped.
- If the column does not contain only those values, the process parses the column based on the specified date format or user exit, if specified, and then examines the values for each unit of the format. Based on site-specific definitions, the value may be skipped. Typical skipped dates include 0000/00/00 and 99992/31. Check with site management for a list of skipped values.

Examples

The following statement ages a date value by one year:

```
AGING (YEARS 1)
```

The following statement ages a date value to the explicit date of January 1, 2008:

```
AGING (EXP_DATE 2008/01/01)
```

Chapter 7. Definition Keyword Parameters

The Optim batch generation process uses the following definition keywords when it creates certain Batch Utility jobs for execution:

- ACCESS_DEFINITION_DEFINE
- COMPARE_DEFINITION_DEFINE
- TABLE_MAP_DEFINE.

This section describes the parameters that make up these definition keywords.

If you create a Batch Utility job outside of the Optim batch generation process, you can use the definition keyword parameters. However, the best practice is to use the Batch Utility keyword to refer to a named definition in the Optim Directory (that is, ACCESS_DEFINITION, COMPARE_DEFINITION, and TABLE_MAP).

ACCESS_DEFINITION_DEFINE Parameters

You can use the ACCESS_DEFINITION_DEFINE keyword in the ARCHIVE and EXTRACT Batch Utility control statements. The parameters must be enclosed within parentheses, with a semicolon after the close parenthesis.

```
[ DESCRIPTION //description// ]
[ SECURITY { PUBLIC | READONLY | PRIVATE } ]
DEFCID defaultcreatorid  START starttable
OWNER ownerid
MODIFIED yyyy-mm-dd-hh.mm.ss
ADDTBLS { Y | N }  MODCRIT { Y | N } BEGINDISP { D | S | A }
ADCHGS { T | P }  USENEW { Y | N } APPLYSELFREF { Y | N }
[ EXPIREVALUE { ddddd | yyyy-mm-dd } ]
[ ROWLIST dsname ]
[ GRPCOL column  GRPROWS n GRPVALS n ]
TABLE ( table  ACCESS { S | U | I | D } REF { Y | N }
   [ EXTFREQ n ]  [ EXTRLIMIT n ] PREDOP { A | O }
   [ SQL //sqlwhereclause// ]  [CORRELNAME correlname ]
   COLFLAG { Y | N }
   [ARCCRIT_IN_WHERE { Y | N } ]
   [ ARCIXTAB { Y | N } ]
   [ ARCDAA { Y | N } ]
   [ ARCOP { A | O } ]
   [ ARC_ACTION
   ( ( type_of_action1 share_status1 [ //sql_statement// ] )
   ( type_of_actionn share_statusn [ //sql_statement// ] ) ) ]
   [ COLUMN ( column  DISP { Y | N } ACCESS { S | U }
      HEADING { N | L } { L | R | C }
      [ SORT n { A | D } ]
      [ PRED //selectioncriteria// ]
      [ ARCIXCOL { D | N | S } ]
      [ ARCFMT 'formatparam' ]
      [ ARCDATE 'explicitdate' ]
      [ ARCPIVOT n ]
      [ ARCYRS  n ]
      [ ARCMOS  n ]
      [ ARCWKS  n ]
      [ ARCDAYS  n ]  ) ]
   [ LEGDSN dsname ]
   [ LEGTYPE { D | I }]
   [ LEGPSB psbname ]
   [ LEGPCB pcbnum ]
   [ LEGSEG segname ]  )
```

```
REL ( relationship  STATUS { NEW | SEL | UNSEL | UNK }
   Q1 { Y | N }  Q2 { Y | N }
   CHILDLIMIT { n | NOCHILDREN | UNLIMITED }
   TYPE { DB2 | OPT | IMS }  DUPRELS { Y | N }
   [PAR_ACCESS { K | S }]
   [PAR_KEYLIMIT n]
   [CHI_ACCESS { K | S }]
   [CHI_KEYLIMIT n]
   PAR parent  CHI child  )
SUBVAR ( :variablename  VALUELEN  101
   [ VALUE // value // ]  [ PROMPT  // prompt // ] )
```

DESCRIPTION
The optional description. This up to 40-character description must be enclosed in double slashes.

SECURITY
The security assigned to the object, specified as:

PUBLIC
All can use and modify (default).

READONLY
All can use, but only owner can modify.

PRIVATE
Only owner can use and modify.

DEFCID
The default Creator ID.

START
The name of Start Table. The Creator ID is included only when it differs from the default Creator ID.

OWNER
The owner of the exported object.

MODIFIED
The date and time the object was last modified.

The following parameters correspond to those specified on the Access Definition Parameters panel.

ADDTBLS
Dynamically add tables. Specify:

Y Users can dynamically add tables during an edit or browse session.

N Users cannot add tables.

MODCRIT
Modify selection criteria. Specify:

Y Users can modify selection criteria during an edit or browse session.

N Users cannot modify selection criteria.

BEGINDISP
The initial display when an Access Definition is used to edit or browse. Specify:

D Data.

S Selection criteria prompt for the Start Table.

A Selection criteria prompt for each accessed table, including the Start Table.

ADCHGS
Save changes made to an Access Definition during edit or browse. Specify:

T (Temporary) Modifications made to an Access Definition during an edit or browse session are used for that session only.

P (Permanent) Modifications made to an Access Definition during an edit or browse session are saved.

USENEW
Use new relationships. Specify:

Y Select new relationships for traversal.

N Do not select new relationships.

APPLYSELFREF
Apply selection criteria when a table is self-referenced. This setting applies only when browsing or editing data. Specify:

Y Apply any criteria for a table each time the table is referenced (default).

N Apply criteria to the Start Table the first time it is accessed, but not when the table is self⌂referenced.

EXPIREVALUE
The optional expiration parameters for Archive Files generated in Archive Processes that use the Access Definition. (Archive only) Specify:

blank No expiration date.

yyyy-mm-dd
An explicit date, in ISO format, after which the data in the Archive File is considered to be expired. An error occurs if an Archive Process uses the Archive File created with this Access Definition after the specified date.

ddddd The number of days until expiration.

The following parameters are included only if a Row List or group selection processing has been defined for the Start Table.

ROWLIST
The name of the data set containing the Row List from a Point-and-Shoot session.

GRPCOL
The name of the column specified for group selection processing.

GRPROWS
The maximum number of rows selected for each unique column value for group selection processing.

GRPVALS
The maximum number of unique column values for group selection processing.

TABLE
A TABLE parameter is provided for each table in the Access Definition. The set of keywords for each table is enclosed in parentheses.

ACCESS
Access rights for the table, specified as:

S Select only.

U Update or select.

I Insert, update, or select.

D Delete, insert, update, or select (default).

REF Reference table identifier.

Y Reference table.

N Not a reference table.

EXTFREQ
An optional numeric selection factor for rows in the table.

EXTRLIMIT
The optional maximum number of rows extracted or archived. Values range from 1 to 4,294,967,295

PREDOP
The logical operator to apply to selection criteria.

A Combine criteria with AND operator (default).

O Combine criteria with OR operator.

SQL Optional SQL WHERE clause text.

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

CORRELNAME
Correlation name specified on the SQL WHERE Clause panel, if applicable.

COLFLAG
Modifications to the Define Columns panel.

Y Specifications on the Define Columns panel were modified.

N The panel was not modified.

ARCCRIT_IN_WHERE
Archive date criteria specified. (Archive only)

Y Archive date criteria is specified for one or more columns.

N Archive date criteria is not specified.

ARCIXTAB
Archive index indicator. (Archive only)

Y One or more columns in the table are indexed.

N Columns are not indexed.

ARCDAA
Delete after archive indicator. (Archive only)

Y Delete source rows after archiving.

N Retain rows after archiving.

ARCOP
The logical operator to apply to archive criteria. (Archive only)

A Combine criteria with AND operator (default).

O Combine criteria with OR operator.

ARC_ACTION
Archive Action definition. An entry is required for each active Action associated with the Access Definition. Each entry may include the following three qualifiers, which are positional and are separated from each other by blanks (not commas).

type_of_action
The type of Action, specified as one of the following (required):

- BEFORE_EXT_1ST_ROW
- BEFORE_EXT_ROW
- AFTER_EXT_ROW
- AFTER_EXT_LAST_ROW
- BEFORE_DEL_1ST_ROW
- BEFORE_DEL_ROW
- AFTER_DEL_ROW
- AFTER_DEL_LAST_ROW
- BEFORE_REST_1ST_ROW
- BEFORE_REST_ROW
- AFTER_REST_ROW
- AFTER_REST_LAST_ROW

share_status
The shared status for the Action, specified as either NOT_SHARED or OWNER (required).

//sql_statement//
The SQL statement for the Action (optional).

Note: An SQL WHERE clause must be delimited by double slashes, "//". If the text exceeds the length of one line, continuation is indicated by slashes at the end and at the beginning of the continuation line. The text can be broken anywhere since the continuation lines are appended without inserting blanks. You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

COLUMN
A COLUMN parameter is provided for every column in a table only when the defaults have been modified for the Access Definition. The set of keywords is enclosed in parentheses.

DISP Indicator for display of column.

Y Display the column (default).

N Do not display the column.

ACCESS
Edit access to a column.

U Column can be updated.

S Column is display only.

HEADING
The column heading, displayed as:

N The column name (default).

L The label specified in the DB2 Catalog. And the position of the heading in relation to the data length for a column, specified as:

L Left-justified

R Right-justified

C Centered (default)

SORT The sort level as a numeric value, and the sort direction, as:

A Ascending (default)

D Descending

PRED

The text of the selection criteria. (Included only if selection criteria are specified.)

ARCIXCOL

Archive index indicator for a column. (Archive only)

D Values in the column have a dense index.

N Values in the column are not index (default).

S Values in the column have a sparse index.

ARCFMT

The format of the date in the column. The format must be appropriate for the data type and precision of the column. (Archive only)

ARCDATE

The cut-off date for archived data, specified in ISO, European, or USA format. Archive converts the date to your DB2 default format. Rows with a date earlier than the specified date are selected for archiving. (Archive only)

ARCPIVOT

The year used to determine whether a two-digit year value is handled as a 20th century (1900) or 21st century (2000) value. If the year is greater than or equal to this value, 1900 is assumed. (Archive only)

ARCYRS

The number of years. This value is combined with any values for months, weeks, and days, and is subtracted from the current date to determine the cut-off date for archived data. (Archive only)

ARCMOS

The number of months. This value is combined with any values for years, weeks, and days, and is subtracted from the current date to determine the cut-off date for archived data. (Archive only)

ARCWKS

The number of weeks. This value is combined with any values for years, months, and days, and is subtracted from the current date to determine the cut-off date for archived data. (Archive only)

ARCDAYS

The number of days. This value is combined with any values for years, months, and weeks, and is subtracted from the current date to determine the cut-off date for archived data. (Archive only)

LEGDSN (for Legacy Table only)

The name of the legacy source data set.

Note: If an IMS database data set is the source data set for this legacy table, and you are using dynamic allocation, specify the *dsname* as '$MDA'. You can omit the LEGDSN parameter if the site option `Require IMS Data set name` is set to the value `N`.

LEGTYPE (for Legacy Table only)

The type of legacy source file.

D VSAM or Sequential file

I IMS file

LEGPSB (for IMS Legacy Table only)

The name of the PSB that provides access to the IMS services that Optim Legacy requires to access the database records. (Included only if LEGTYPE is I.)

LEGPCB (for IMS Legacy Table only)
The relative number of the database PCB within the specified PSB that grants Optim Legacy the authorization to manipulate the data. (Included only if LEGTYPE is I.)

LEGSEG (for IMS Legacy Table only)
The names of the segments within the specified DBD. (Included only if LEGTYPE is I.)

REL A REL parameter is provided for each relationship in the Access Definition, as listed on the Specify Relationship Usage panel. The set of keywords for each relationship is enclosed in parentheses.

STATUS
The status of the relationship, specified as:

NEW New to the relationship list.

SEL Selected.

UNSEL
Not selected.

UNK Not defined for the listed tables.

Q1 Value specified for Question 1.

Y Process all parent rows (default).

N Do not process all parent rows.

Q2 Value specified for Question 2.

Y Process all child rows.

N Do not process all child rows (default).

CHILDLIMIT
The maximum number of child rows processed for each parent row, specified as:

n An explicit number of rows.

NOCHILDREN
No child rows.

UNLIMITED
All related child rows.

TYPE The type of relationship, specified as DB2, OPT (Optim), or IMS.

DUPRELS
Indicator for duplicate relationships.

Y At least one other relationship with the same parent and child tables.

N No duplicate relationships.

PAR_ACCESS
Access method used to retrieve rows from the parent table when the traversal is from child to parent. Optim determines the default if this keyword is omitted.

K Access by key lookup.

S Access using table scan.

PAR_KEYLIMIT
Maximum number of parent table rows that can be retrieved in one SQL call, when the traversal is from child to parent. Specify a value in the range 1 - 100. The default is 1. This parameter applies only if PAR_ACCESS is K.

CHI_ACCESS

Access method used to retrieve child table rows when the traversal is from parent to child. Optim determines the default if this keyword is omitted.

K Access by key lookup.

S Access using table scan.

CHI_KEYLIMIT

Maximum number of child table rows that can be retrieved in one SQL call, when the traversal is from parent to child. Specify a value in the range 1 - 100. The default is 1. This parameter applies only if CHI_ACCESS is K.

PAR The name of the parent table in the relationship.

CHI The name of the child table in the relationship.

SUBVAR

A SUBVAR parameter is provided for each substitution variable that has been defined in the Access Definition. The set of keywords for each substitution variable is enclosed in parentheses. The substitution variable name must be preceded by a colon (:), and can be up to 12 characters.

VALUELEN

The value length is always 101. This keyword must immediately follow the substitution variable name.

VALUE

The optional default value for the substitution variable. This up to 100-character value must be enclosed in double slashes.

PROMPT

The optional prompt text to be displayed for the substitution variable. This up to 35-character text must be enclosed in double slashes.

TABLE_MAP_DEFINE Parameters

You can use the TABLE_MAP_DEFINE keyword parameters in the INSERT, COMPARE, CONVERT, and RESTORE Batch Utility control statements.

The parameters must be enclosed within parentheses, with a semicolon after the close parenthesis.

```
[DESCRIPTION //description//]
[SECURITY {PUBLIC | READONLY | PRIVATE}]
OWNER ownerid
MODIFIED yyyy-mm-dd-hh.mm.ss
VALRULES { C | M }
SRCCID sourceid  DESTCID destid  [COLMAPID colmapid]
{SRCEXT dsname | SRCAD ad}
{DESTEXT dsname | DESTAD ad}
[SRCTYPE { X | A | T }] [DESTTYPE { X | A | T }]
    (sourcetablename=desttablename
      [ {CM mapname | LOCALCM (mapdef)}]
[LEGTYPE { D | I }] [LEGDSN dsname]
[LEGPSB psbname] [LEGPCB pcbnum] [LEGSEG segname]
[LEGEXIT exitname]
[ARC_ACTION
        ((type_of_action1 share_status1 [//sql_statement//])
        (type_of_actionn share_statusn [//sql_statement//]))]
[PROCESSMODE { I | U | B } ] [DELETEFLAG{ Y | N }]
[MATCH_KEY name]   [USER_MATCH_KEY name]
[ACCESS_METHOD { K | S } [KEY_LIMIT  n]
[MATCH_KEY_PROPERTY value] [MATCH_METHOD value]
[SOFT_MATCH_ENABLED { Y | N } ][PERCENT_REQUIRED percentage]
[PERCENT_ACCEPTABLE percentage] [MAXIMUM_ATTEMPTS  n]
```

DESCRIPTION
The optional description. This up to 40-character description must be enclosed in double slashes.

SECURITY
The security assigned to the object, specified as:

PUBLIC
All can use and modify (default).

READONLY
All can use, but only owner can modify.

PRIVATE
Only owner can use and modify.

OWNER
The owner of the object.

MODIFIED
The date and time the object was last modified.

VALRULES
The validation rules for the Table Map, specified as:

M	Move or Archive rules.
C	Compare rules.

SRCCID
The default Creator ID for the source tables.

DESTCID
The default Creator ID for the destination tables.

COLMAPID
The optional default Column Map ID.

The following parameters are included, depending on the source and destination types, as well as whether the Table Map is for Move, Archive, or Compare.

SRCEXT
The name of the data set containing an Extract or Archive File used as the source.

SRCAD
The fully qualified name of the Access Definition used as the source.

DESTEXT
The name of the data set containing an Extract or Archive File used as the destination.

DESTAD
The fully qualified name of the Access Definition used as the destination.

SRCTYPE
The source type as:

X	Extract or Archive File
A	Access Definition
T	Table

DESTTYPE
The destination type as:

X	Extract or Archive File
A	Access Definition

T Table

Table Mapping
The following source table to destination table mapping information is provided for each pair of tables referenced in the Table Map. At least one pair of tables must be specified.

sourcetablename
The source table name. The Creator ID is included if it differs from the SRCCID.

desttablename
The destination table name. The Creator ID is included if it differs from the DESTCID. The words "NOT-SPECIFIED" are inserted when the destination table in a pair is omitted.

CM or LOCALCM is included if a Column Map has been specified for a pair of tables. ARC_ACTION is included if any Archive Actions apply.

CM The name of the Column Map for the pair of tables. The Map ID is included if it differs from the COLMAPID.

LOCALCM
The local Column Map definition, enclosed in parentheses. Only the source and destination mapping (*source-expr* = *dest-column*) is included here. For details, see Export and Import Optim Objects in the *Common Elements Manual*.

The following parameters, all named with "LEG" prefix, apply only to Legacy Tables referenced in the Table Map. The type of legacy source file determines which parameters are included. (Optim Legacy only)

LEGTYPE
The type of legacy source file to be used with the Legacy Table:

D VSAM or Sequential file

I IMS file

LEGDSN
The name of the destination legacy data set.

Note: If an IMS database data set is the source data set for this legacy table, and you are using dynamic allocation, specify the *dsname* as '$MDA'. You can omit the LEGDSN parameter if the site option `Require IMS Data set name` is set to the value `N`.

LEGPSB
The name of the PSB that provides access to the IMS services that Optim Legacy requires to access the database records. (Included only if LEGTYPE=I)

LEGPCB
The relative number of the database PCB within the specified PSB that grants Optim Legacy the authorization to manipulate the data.

LEGSEG
The names of the segments within the specified DBD. (Included only if LEGTYPE=I)

Note: Additionally, a Table Map referencing one or more Legacy Tables for IMS data includes parameters for any Retrieval Definitions that are referenced by the Legacy Table. (For details, see Export and Import Optim Objects in the *Common Elements Manual*.)

LEGEXIT
The name of the I/O Exit Load module (if different from the default value). This exit will be called only if the appropriate Enable I/O Exit site option is U. Otherwise it will be ignored.

ARC_ACTION

Archive Action definition. An entry is required for each Archive Action associated with the Table Map. Each entry may include the following three qualifiers, which are positional and are separated from each other by blanks (not commas).

type_of_action

The type of Action, specified as one of the following (required):

BEFORE_REST_1ST_ROW

BEFORE_REST_ROW

AFTER_REST_ROW

AFTER_REST_LAST_ROW

share_status

The shared status for the Action, specified as either NOT_SHARED or OWNER (required).

//*sql_statement*//

The SQL statement for the Action (optional).

Note: You must leave a space after a comma that precedes a numeric value if the DB2 setup specifies a comma as the decimal point value.

The following parameters are included only if the Table Map contains processing overrides.

PROCESSMODE

The processing mode for the table.

I Insert only.

U Update only.

B Both insert and update.

DELETEFLAG

Delete parameter for the table.

Y Delete all rows in the destination table prior to inserting data.

N Do not delete.

MATCH_KEY

For a Compare process only. Name of the match key used for this comparison.

USER_MATCH_KEY

For a Compare process only. Name of the user-defined match key for this comparison.

ACCESS_METHOD

The method used to access the parent or child table for each relationship. Specify:

K Key lookup locates rows using a WHERE clause to search for primary or foreign key values. Additionally, you can change the maximum number of key lookups performed at one time for a table.

S A scan reads every row in a table at one time.

KEY_LIMIT

Value for the maximum number of key lookups performed at one time for a table. This value is used whenever a key lookup is the access method used to scan a table and no specific key lookup limit is specified. Specify a value in the range 1 to 100.

MATCH_KEY_PROPERTY

Indicates the presence of a match key for the sources to be compared. The value in this field is automatically populated, based upon the known attributes of the data. If this value is Unique, the remaining fields on this panel are non-modifiable. Specify one of these values:

Unique
Unique

M Multiple, non-unique.

N Not keyed. Data has no key defined.

MATCH_METHOD
Controls the processing for non-uniquely keyed and non-keyed sources. This field is not modifiable and displays the value ANY.

SOFT_MATCH_ENABLED
This parameter is valid only if MATCH_METHOD is A.

PERCENT_REQUIRED
This parameter is valid only if SOFT_MATCH_ENABLED is Y. Specifies the percentage of similarity required to consider two unequal rows matched. This value applies only to the columns that are not defined as part of the match key. Allowable values range from 1 through 100. The default is 50 percent.

PERCENT_ACCEPTABLE
This parameter is valid only if SOFT_MATCH_ENABLED is Y. Allowable minimum value is the value specified for PERCENT_REQUIRED; allowable maximum is 100. The default is 100.

MAXIMUM_ATTEMPTS
This parameter is valid only if SOFT_MATCH_ENABLED is Y. Limits the number of compares performed for any set of unequal rows. The value 0 disables this limit. Specify a value in the range 1 to 999,999. The default is 0.

COMPARE_DEFINITION_DEFINE Parameters

You can use the COMPARE_DEFINITION_DEFINE keyword parameters in the COMPARE Batch Utility control statement.

The parameters must be enclosed within parentheses, with a semicolon after the close parenthesis.

```
[ DESCRIPTION //description// ]
[ SECURITY PUBLIC | READONLY | PRIVATE ]
OWNER ownerid
MODIFIED yyyy-mm-dd-hh.mm.ss
SRC1TYP { EXTR dsname | AD ad | LOCALAD ( addef ) }
SRC2TYP { EXTR dsname | AD ad | LOCALAD ( addef ) | ALL }
TM ( tablemap )
[ REL ( relname STATUS { SEL | UNSEL | REF }  SOURCE { 1 | 2 }
   TYPE { DB2 | OPT }  PAR parent CHI child ) ]
[ CMPDSN14 dsname ]
```

DESCRIPTION
The optional description. This up to 40-character description must be enclosed in double slashes.

SECURITY
The security assigned to the object, specified as:

PUBLIC
All can use and modify (default).

READONLY
All can use, but only owner can modify.

PRIVATE
Only owner can use and modify.

OWNER
The owner of the object.

MODIFIED
The date and time the object was last modified.

SRC1TYP
The type and name for Source 1, specified as one of the following:

EXTR Fully qualified name of the Extract or Archive File.

AD Fully qualified name of the Access Definition.

LOCALAD
Entire Access Definition specification.

Note: For details on the contents of the Access Definition, see "ACCESS_DEFINITION_DEFINE Parameters" on page 149.

SRC2TYP
The type and name for Source 2, specified as one of the following:

EXTR Fully qualified name of the Extract or Archive File.

AD Fully qualified name of the Access Definition.

LOCALAD
Entire Access Definition specification.

ALL All tables in the database. The tables are listed in the Table Map specifications.

CMPDSN
The name of the data set for the results of the Compare Process. Included only if a data set name was specified.

TM The complete definition of a Table Map, TM, is always included. The parameters are specified within parentheses. For details on the contents of the Table Map, see "TABLE_MAP_DEFINE Parameters" on page 156.

REL A REL parameter is provided for each relationship in the Compare Definition. The set of keywords for each relationship is enclosed in parentheses.

STATUS
The status of the relationship, specified as:

SEL Selected

UNSEL
Not selected

REF For a reference table

SOURCE
Indicates the source of the relationship as Source 1 or 2.

TYPE The type of relationship, specified as DB2 or OPT (Optim).

PAR The fully qualified name of the parent table in the relationship.

CHI The fully qualified name of the child table in the relationship.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator
Director of Engineering, Information Management
111 Campus Drive
Princeton, NJ 08540
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Java™ is a trademark of Sun Microsystems, Inc.

Linux is a registered trademark of Linus Torvalds.

UNIX is a registered trademark of The Open Group.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Other company, product, or service names may be trademarks or service marks of others.

Index

Printed in USA